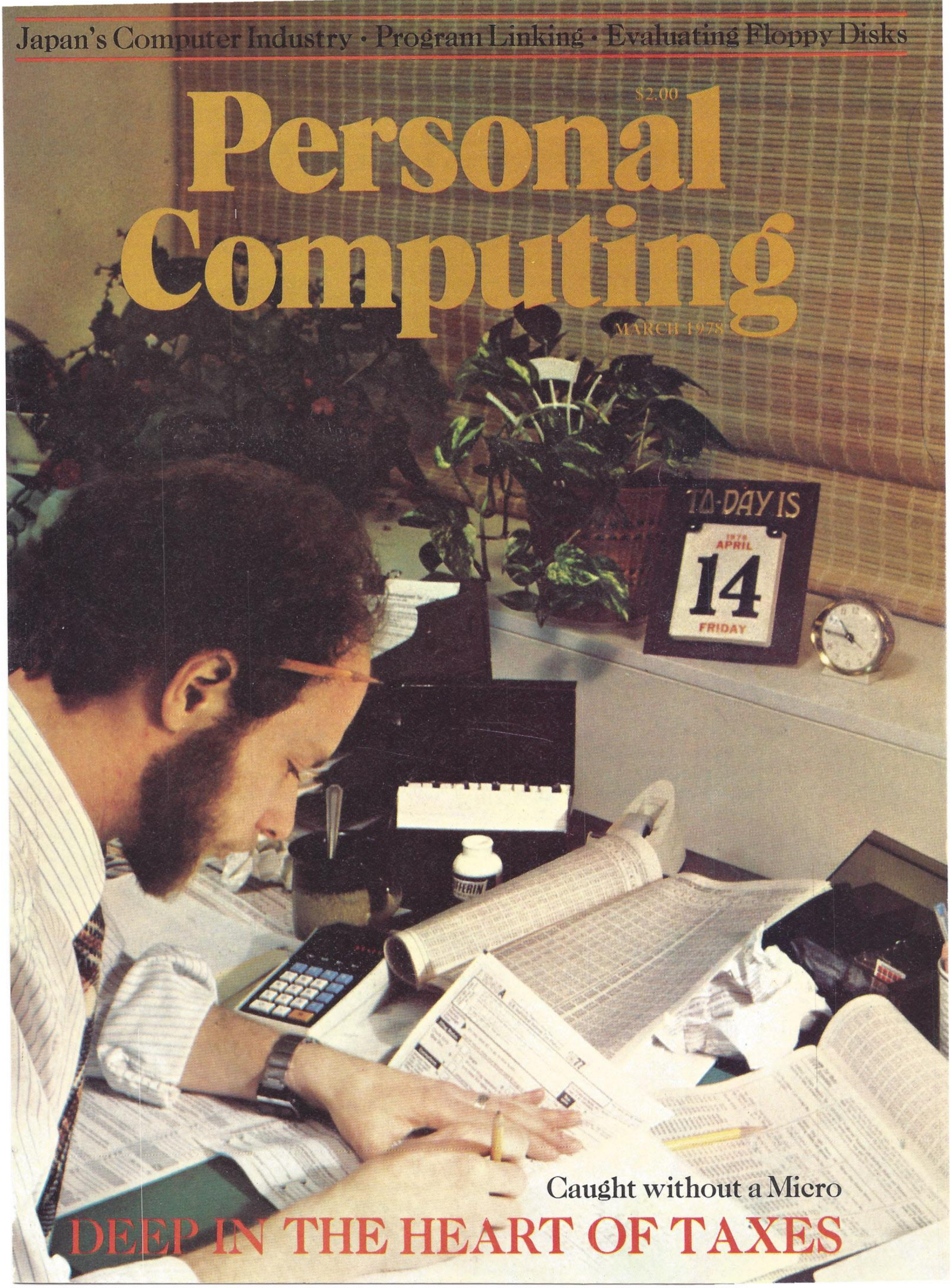
Japan's Computer Industry · Program Linking · Evaluating Floppy Disks

$2.00

Personal Computing

MARCH 1978



Caught without a Micro

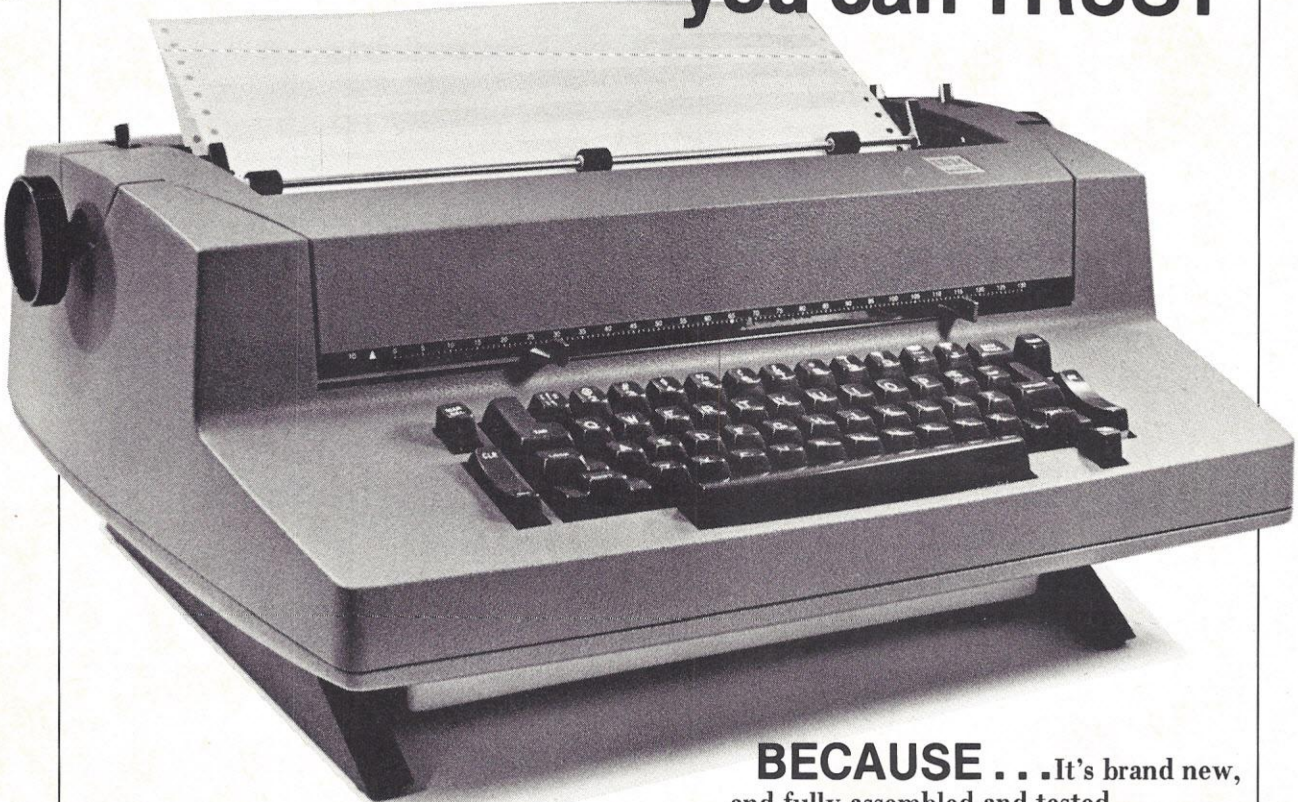
DEEP IN THE HEART OF TAXES

New!

SELECTERM

SYSTEM 9710

The SELECTRIC II* Printer you can TRUST



BECAUSE . . . It's brand new, and fully assembled and tested.

BECAUSE . . . After extensive engineering design and testing by Micro Computer Devices, IBM Corporation has approved the SELECTERM for use with your microcomputer, and provides you with their factory warranty and yearly service agreement for the typewriter. In addition, the electronics conversion portion is fully warranteed by Micro Computer Devices.

BECAUSE . . . You can connect the SELECTERM to your computer within minutes of taking it out of the carton.

IT'S THAT EASY! AND THAT RELIABLE!

FEATURES
- Complete ASCII character set in supplied element.
- Full upper, lower case alphanumeric characters.
- Tab Command, Index (vertical tab), Backspace, Bell—all under computer control.
- Parallel Interface, standard.

ALL ELECTRONICS INCLUDED
- Power supply, electronics and cable sets included to permit immediate connection to the parallel port of any computer, at standard TTL level.

SOFTWARE
- All necessary conversion software in PROM to handle ASCII input, directly.

PRINTER or TYPEWRITER
- May be used as a standard typewriter when not in use with your computer.

OPTIONS
- Dual Pitch, $125
- Correction Feature, $125
- Tractor Feed Platen, $250
- Noise Reduction Feature, $50

AVAILABLE SOON
- RS-232 Interface

PRICE and DELIVERY
- Assembled and tested, $1750
- Available ONLY from authorized dealers.
- Delivery 1 to 2 weeks from receipt of order.
- OEM delivery in quantity within 30 days.

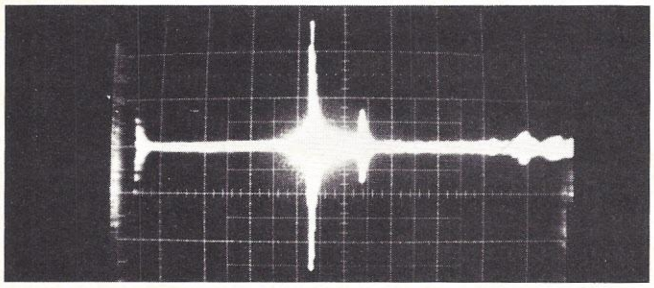
SELECTERM

micro computer devices inc.
960 E. Orangethorpe, Bldg. F
Anaheim, California 92801
Telephone (714) 992-2270

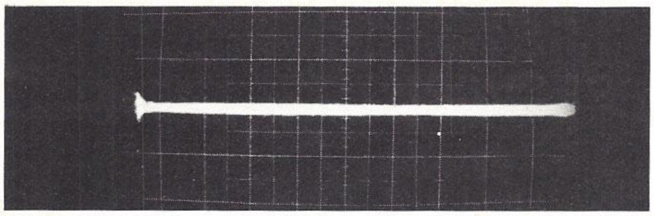
*Registered trademark of IBM Corporation

"Innovators to the Microcomputer Industry"

Did your soldering iron blow your MOS?

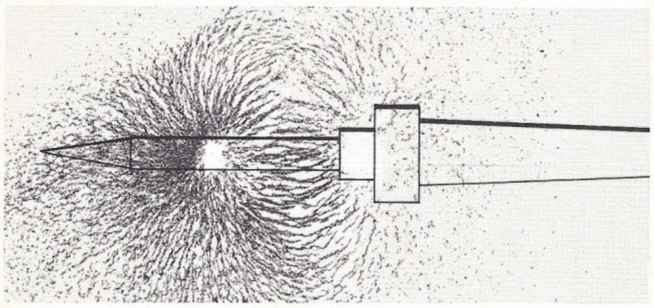


Competitor 27v Spike Scale: 10v/cm

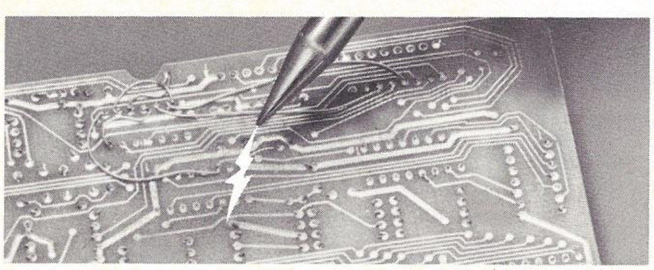


UNGARmatic 3v Spike Scale: 10v/cm

TRANSIENT SPIKES caused by the switching action of some controlled output soldering stations may be transmitted to the workpiece and may adversely affect a metal oxide semiconductor, particularly if the amplitude of that spike is in excess of the operating voltage of the device.



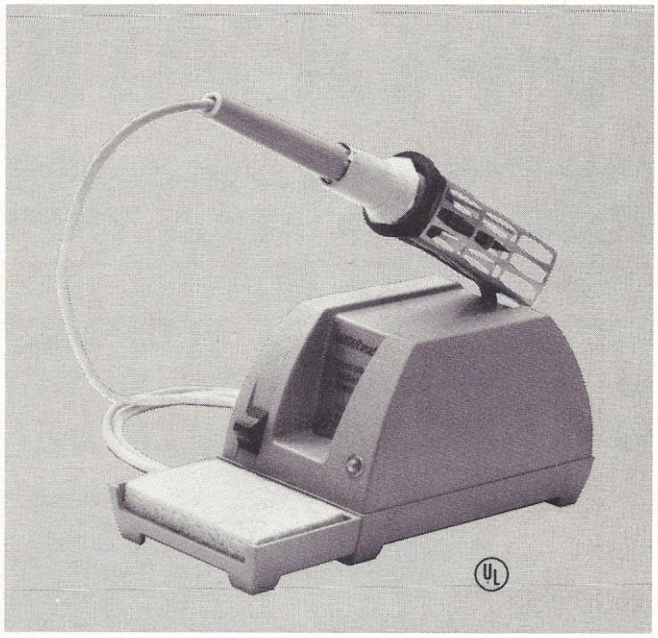
MAGNETIC FLUX surrounds the tip in those controlled output soldering stations which rely upon a magnet and curie effect alloy to control the tip temperature. Such magnetic flux is powerful enough to lift a paper clip, and it can transport ferrous contaminates to a solder connection. When passed over a conductor, the magnet will cause a voltage in that conductor which may have an undesirable effect on the component being soldered or even another component in the same circuit.



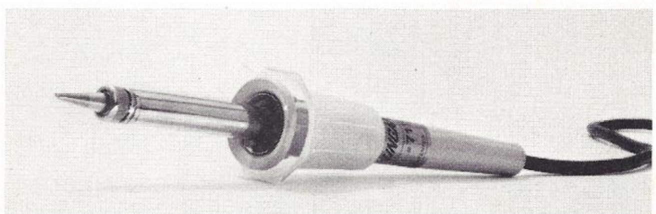
STATIC ELECTRICAL CHARGES on a soldering iron tip may reach levels of thousands of volts unless those charges are drained to earth by adequate grounding of the tip.

STOP blowing your MOS devices

1. Utilize conductive card racks, bench tops and tools which are at the same potential, preferably ground zero.
2. Operators should wear grounded wrist straps.
3. Frequently check the continuity of the ground circuit.
4. Use a soldering iron with a fully grounded tip, such as an UNGARmatic® Controlled Soldering Station.
5. Use a soldering iron which suppresses the transient switching spike to less than the 5 volt operating voltage of sensitive MOS devices. Use an UNGARmatic® Temperature Controlled Soldering Station.



6. Use a soldering iron which accurately controls the tip operating temperature to protect against circuit overheat damage yet maintains the proper temperature for repeatable, high quality solder connections. Use an UNGARmatic® Controlled Soldering Iron.



7. Use a soldering iron which may be custom tailored to your application, yet one which is economical to purchase and economical to maintain. Use a thread-together modular UNGARmatic® Temperature Controlled Soldering Iron.

Designed for Professionals by Professionals

Just call or write us for more information.

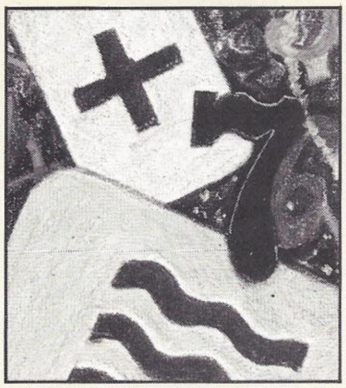
Ungar.

Division of Eldon Industries, Inc.
P.O. Box 6005, Compton, Ca. 90220 — (213) 774-5950
In Canada, 50 Prince Andrews Place, Don Mills, Ontario.

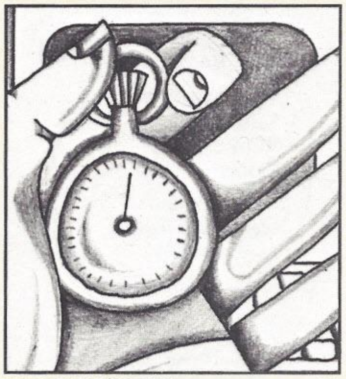
CIRCLE 4

March 1978

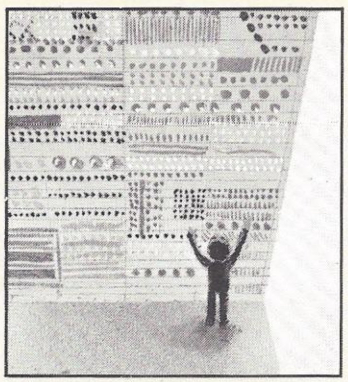
VOLUME II, NO. 3



TESTING ESP 19



REACTION TIME 53



LOOKING BACK 51

Cover photograph by Jon Buchbinder

DEPARTMENTS

LAUNCHING PAD

ON THE LIGHTER SIDE

PERSONAL COMPUTING INTERVIEW

DIGGING IN

ONCE UPON A TIME

Personal Computing

MARCH 1978 VOL. II, NO. 3

Subscription rates. U.S.: 1 year (12 issues) $14; 2 years (24 issues) $26; 3 years (36 issues) $38. Canada & Mexico: add $4/year for surface mail, $8/year for airmail. In Japan/Asia: Personal Computing, c/o CQ Publishing Co., 14-2 Sugamo 1-chome, Toshima-ku, Tokyo 170, Japan. All other countries: add $8/year for surface mail, $36/year for airmail. Send subscription orders and changes of address to: Circulation, Personal Computing, 1050 Commonwealth Ave., Boston, MA 02215. **Back issues.** U.S.: $3. Canada & Mexico: $4. All other countries: $6.

Publisher. Published monthly by Benwill Publishing Corp.: Harold G. Buchbinder, Chairman of the Board; George Palken, President; Esther Shershow, Treasurer. Executive, Editorial and Subscription Offices: 1050 Commonwealth Ave., Boston, MA 02215. Controlled Circulation postage paid at Long Prairie, MN. Membership in Audit Bureau of Circulation pending.

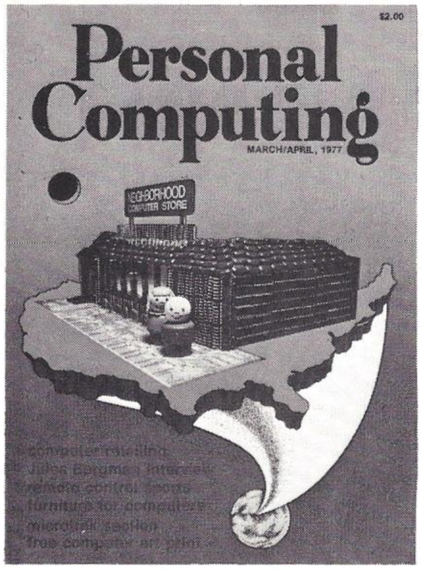
Star burst

Editors:
As a charter subscriber, I'd like to say you're doing a fine job exposing the personal computing world to both the initiated and the uninitiated personal computer user. Having been involved in one of the first operational military solid-state computer operations (POLARIS Navigation – 1960), I consider myself somewhat of an old hand – but I am still learning!

This letter is prompted by the article "A Star Trek Walkthrough"(March/April). I finally screwed up my courage, and, armed with a magnifying glass, translated the L.E. Cochran program into a BASIC dialect suitable for my SWTPC 6800. (I admire Mr. Cochran's ingenuity in slipping his name into the program. He made it essential to the running of the system (line #200) by printing only the left 8 characters of "STARTREK ADAPTED BY L.E. COCHRAN 2/29/76" as the title for the introduction to the game.)

There are several errors in the Star Trek printout. The most glaring is a tendency to print "-" in lieu of "+". This occurs in statements 130, 160 (twice) and 190. The statements should read: K9=K9+K, B9=B9+B, Q(I,J)=K*C+B*W+INT(RND(Y)*X+Y), and Q(X,Y)=Q(X,Y)+10. As written, the number of Klingons and number of starbases is negative, and the quadrant map by sectors (short-range sensors) does not match the data in the galactic map/long-range sensors. Further, only if you blow up a starbase with your photon torpedo do you get a "Good Work!" accolade – which is either an error or supreme sarcasm!

Mr. Cochran's programming techniques are efficient in minimizing use of random access memory. He has interwoven torpedo function into warp function so the warp course statements will apply to the torpedo course within a quadrant. He also uses variable names many times to store different quantities – this is fine for memory saving, but makes program tracing much more difficult! E$ is variously "ENTERPRISE FROM", "KLINGON AT",



"YES" and "NO". It comes out even in the end but it takes a lot of time.

A confusing factor in the Star Trek program is the use of the IF THEN statement when there are multiple statements to a line. The IF condition applies to all statements to the right of IF. To accomplish this in SWTPC BASIC requires a sequence as follows:

```
250  IF A=0 THEN 252
251  GOTO 260
252  A=2
253  GOTO 1050
```

The SWTPC BASIC will not allow branching out of a FOR-NEXT loop before at least once past the NEXT statement; hence a true condition for line 930 results in a halt and error message when occurring on the first iteration. This is corrected by computing the destination at line 890, and, if outside the initial quadrant, branching to line 1110 before entering the FOR-NEXT loop.

Once all these adjustments are made, the program runs beautifully on my SWTPC 6800. The ENTERPRISE stops when its path is blocked by a star, the sensor and galactic records reflect every destruction of a Klingon, starbase or a star, and the spacestorms regularly disable various systems.

Robert Irving
Northridge, CA

Pet Problems

Dear Editor:
I have had my PET for about a month now and thought I'd write down some of my observations about it. Basically, I'm very pleased with it but there are things I'd like to see improved.

For educational use there should be a composite video signal coming out of a UHF or BNC Connector (not the oddball one that Radio Shack has used). This permits the teacher to bring the PET into the classroom to work with his entire class.

The PET should also have a handle on it. One of the first things I did after receiving mine was to buy a 99¢ handle at the hardware store. It is very helpful when carrying the PET into class, home or onto an airplane. (You should see the airport security people when I present this strange looking TV set for inspection!)

We've designed and installed a simple 4-channel A/D converter to permit us to enter parameter values into programs during execution of a simulation. It permits the digital computer to act like an analog computer (but with none of the problems of the real analog computer) and permits enormously improved simulation experiences. Parts for this cost about $2.

We have had 6-10 system crashes in the month we've had our PET. Although we haven't identified the cause, I suspect it may be a voltage spike riding in on the line. The system suddenly hangs up and the keyboard goes dead. It has always been easy to recover just by turning the PET off and on, but then the memory is zeroed and the program disappears. Is it possible to restart without zeroing memory? Such a capability would sure be nice.

The monitor doesn't have equal gain horizontally and vertically. As a consequence, when drawing a circle (in a math program) it comes out an ellipse. A square (equal numbers of horizontal and vertical elements) comes out a rectangle. I have found a height adjustment in the monitor but no width control. Is there one? (Inciden-

Continued on next page

tally such controls probably should come out the back.)

It really would be nice to be able to, in one vertical and one horizontal command, send the cursor to any place on the screen (e.g., POKE 245, V). With this, graphing would really be easy. This amounts to a two-dimensional TAB command.

The cassette recorder doesn't have a counter and it doesn't have fast forward. The former I can understand because of price, but the latter I can't. This means I can only put a few programs on a cassette because I must initiate the search at the beginning of the tape. I solve it by writing only a couple of programs onto a cassette, but then I only use 2% of the tape's capacity.

Despite these criticisms I am very happy with the PET. I'm excited about this machine for educational applications because of its powerful BASIC, because of its price, and because of its portability. The latter property is especially exciting because it gives us teachers, for the first time, the opportunity to let our students take a computer home overnight or over a weekend to develop an idea through use of a simulation or to solve a problem normally beyond their capability.

Ludwig Braun
Professor and Asst. Director
for Educational Technology
Stony Brook, NY

Newer Discovery

Dear Editors:

Thanks for the pleasure I had this Christmas morning, playing with your "New Discovery" (Nov/Dec PC).

But, there is a mistake in the page 44 listing: Bet #3 should read: 3-3-6-15-9. Second, you missed a point; if N>4 you are betting $1 too much. You agree? Try a $1 bet on Bet #2: 2-1-2-5-3 etc. . .

So your program should be modified so one can bet less and play a slightly longer game, with a slight increase in investment.

Henri Reiher
Montreal, Quebec

The author replies: On your point #1, you're right, of course. Evidently my mental computer doesn't handle the programming as well as the Compal-80 does.

As for point #2, well – we set the specs for our betting plan as "increase . . . by at least a dollar each time you miss." One can bet less aggressively and last longer; the tradeoff is smaller profit if/when you do hit a winner.

Winging it

Dear Editors:

While reading "Computer Modeling" (Sept., 1977) I found an error.

The author stated that the Wright brothers did not use a wind tunnel, and implied that they did not use air foil equations. This simply is not true. I suggest the author visit Kitty Hawk and examine the air foil equations and wind tunnel used by the Wright brothers.

Both items, used in their first successful flight, are on exhibit in the museum.

Edward Comer
Greensboro, NC

Gill Apparatus

I am embarking on a long term project to produce tactile (Braille) maps on a large scale, chiefly for schools for the blind, as well as for other institutions for the blind and for blind persons in the developing nations.

I am a microbiologist who has just retired 9 years early (at 58) from the University of California in order to serve as the volunteer Clerk of the I.F.B. Consequently, like 99% of my colleagues in my age bracket, I know nothing about computers. But, I want to build a "Gill Apparatus" and need some information on specifications and costs of such equipment. (This is simply a device which allows one to transfer the X and Y coordinates of an ink-print map, plus a vertical component for tactualization, via a minicomputer onto a tape.)

I plan to accumulate this literature, which I hope will have prices in it, and have it evaluated by some of the people here at our University of California Davis campus who work in the computer field. Price is not much of a consideration, since I am sure to be able to secure from generous persons the funds needed to get the equipment for this important job to serve over 50×10^6 persons on this mortal coil who are sightless.

Donald Montgomery Reynolds
Office of the Clerk
International Federation of the Blind
Four Parkside Drive
Davis, CA 95616

Parlez-vous?

I have a particular problem – a need to store whole phrases or sentences in a foreign language for recall by reference to a particular word within the phrase. The computer would be doing nothing that could not be accomplished (with an infinity of labor) by writing down each phrase and filing it alphabetically by the key word.

In learning a language through reading it is best to master new words and idioms without reference to English. If you encounter the word often enough, the various contexts will define it with greater and greater precision. So, we will type each troublesome word as a heading – then type the whole phrase embodying the word. On recall, all the phrases or sentences previously stored for that word would be displayed.

That's the idea. In my case, the language is Greek, which means not only the Greek alphabet, but provisions for the three forms of accent which are essential in the modern language. Can anyone help?

William G. Raoul
538 West Brow Rd.
Lookout Mountain, TN 37350

Kit or conversion?

Could anyone supply me with a means for converting my Selectric to ASCII printer?

If no kit is available, is there somewhere I could obtain complete, specific detailed instructions on how to make this conversion?

Walter C. Rieker
Sycamore & Mill Streets, Box 52
Clifton Heights, PA 19018

Sailing Along

I'm interested in any information available regarding the application of computer systems to Sailboat Racing.

Particularly I'm interested in knowing if there are any programs available to optimize the sailing angles and sail selection, as well as any games that simulate a race and related tactics.

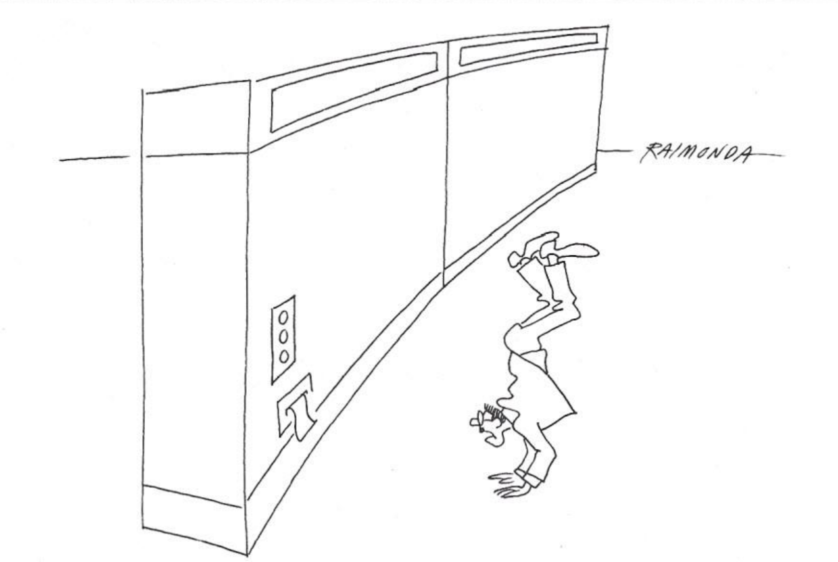
John C. MacLaurin
9263 Third Street
Beverly Hills, CA 90210

Drop them a line

I am now forming a 4-man Family Medical Group and am interested in buying one of the new "microcomputers" for use in keeping medical records, processing insurance claim forms, and keeping financial records and financial analyses of my clinic operation, as well as real estate ventures.

I'm looking for companies manufacturing a unit which might fill the requirements for this type of operation.

Robert T. Cates, M.D.
120 Chinquipin Cove
Jackson, MS 39211



Yeah, but can you do this?


ANNOUNCING!
Software Development
Systems by SPACE BYTE™
SPACE BYTE Software Development Systems are versatile and cost effective; featuring the SPACE BYTE 8085 CPU, self contained computer, 16K SPACE BYTE fully static RAM and the new SPACE BYTE 2708/2716 EPROM Programmer. All PROM programming, I/O handling FDOS, and complete debugging routines, with breakpoints are intrinsic to the SPACE BYTE system monitor. Complete software support too, with iCOM FDOS III, 8080, 8085, Z-80 relocatable macro assembler, operating system, Disk Extended Basic, CP/M and FORTRAN-80 and the SPACE BYTE exclusive 3870/F8 cross assembler.
THE SPACE BYTE SYSTEMS GROUP
Software Development Systems
Business Systems
Word Processors
SPACE BYTE MICRO COMPUTER PRODUCTS ARE SUPPORTED BY COMPUTER RETAILERS WORLD WIDE.
1720 Pontius Ave. Suite 201
Los Angeles, CA 90025 · (213) 468-8080
SPACE BYTE SOFTWARE DEVELOPMENT SYSTEM PRICING STARTS AT $5900.
CPU · RAM
EPROM
PROGRAMMER
SPACE BYTE
SPACE BYTE

RANDOM ACCESS

Sheer Luck, Holmes?

With Kojak, Baretta, and Barnaby Jones limited to one hour of crime-solving a week, trained personnel using traditional, costly and time-consuming manual searching and matching procedures for fingerprint identification have plenty of work to keep them busy. Inflation and the severe competition for budget dollars also reinforce the need for high speed, automated methods for accurate fingerprint matching and identification.

Both automated and semi-automated computer-aided equipment capable of rapid fingerprint identification do exist. Individual fingerprints can be identified by their endings and ridge branches or minutiae (characterized by location, X and Y, and direction, theta, in the Cartesian coordinate system of the three dimensional position identification).

One such system, Fingermatch, by Calspan Technology Products, Inc., incorporates a control console, a latent (crime-scene) information terminal, a lifter position

encoding pen, and one or more processor/memory units.

In operation, a latent fingerprint is placed over the optical system aperture which projects the magnified fingerprint image onto a ground glass surface on the lifter console, roughly centered over a grid.

To encode each minutiae, the operator places the position encoding pen at the tip of a ridge or valley ending, touching the pen to the screen surface. After following the ridge or valley away from its ending for about ½ inch, the operator lifts the pen. The system measures the position and angle of the minutiae by sensing the position of the depressed pen. The system forwards this information, in digital form, to the processor for matching with stored records.

But while fully automated systems are available to digitize every aspect of individual inked prints, the encoding of fainter and more degraded latent fingerprints still requires interpretation by human fingerprint technician or examiners.

It's elementary, my dear Kojak.

Increasing your restaurant's revenue

You may not be personally familiar with the rising cost of operating a restaurant, but we're all more than aware of the increasing figures on the right hand side of the menu.

To help restaurant owners control food costs, raise labor productivity and analyze profit contribution, a Texas computer system company has put a desk computer to work.

Small Business Computer Service (SBCS) of Austin combines an HP 9896 Business Information Management System and accompanying software programs with their own specially designed food handling and record keeping programs to form an easy-to-operate computer for restaurants.

Restaurant owners use the system for general ledger, payroll, accounts payable and receivable, inventory, and recipe, menu cost, labor and sales analysis. Operation of the system requires no knowledge of computers.

The recipe cost analysis program looks at recipe costs from three different perspectives: the actual cost of each item, the average cost of all items with the same commonality code and the highest of all items with the same code. Through this program, the computer can selectively search the data base and print a complete listing of costs per recipe divided by ingredient part number and description, measure and type of unit used, quantity, ingredient cost and total cost of all ingredients.

Similar to recipe cost, menu cost analysis provides a list of ingredients and costing information. In addition, this program provides data on labor and operating expenses, desired profit, suggested and actual selling price, and raw and net raw food cost catalogued according to menu code, name, selling price and cost.

Both menu recipe cost analysis programs derive their information from the computer's memory of inventory data. By updating information daily, the owner receives warnings of developing trends in sales, changing profit margins and variable costs necessary to make adjustments to the restaurant's operations.

With the labor analysis program, the computer acts as a time clock for employees, calculates

tip discounts and tip credits, compiles a record of all payments and prints the employee payroll checks.

For the owner, the computer prints a daily labor overview summarizing sales per hour worked, food and bar customer count, food and bar sales, and average food and bar check. The computer also combines this data to produce a productivity index — minutes worked per customer served.

According to Don Vivian, president of SBCS, one Texas restaurant owner estimates he has reduced labor costs $500 per week using the computer — which isn't bad, considering the system can be leased for less than $500 a month.

Now, if only the price of a nine-course French dinner would go down . . .

Hollywood Debut

If you've been too busy watching your CRT you may have missed the computer's entrance into Hollywood.

In a recent episode of TV's "Welcome Back Kotter", a computer terminal takes Kotter's place in the classroom — and fails. The entire class falls asleep.

One of the more popular TV police series included a scene featuring a Pet look-alike computer being sold at the racetrack from the back of a station wagon — preprogrammed with betting programs . . .

On "Saturday Night Live", a computer made its debut (and concurrent failure) in a skit with a sub-plot about computer dating.

And the list goes on and on. But strangely enough, none of the shows (except where larger police computers are concerned) have shown successful use of computers to accomplish tasks. TV moguls seem to prefer making fun of mechanical limitations and quirks. Next Norman Lear will be producing "8080, 8080" or "Silicon Valley Tonight" . . .

In The Reel World

You think you have headaches with your empties? How about empties that measure from 50 to 84 inches in diameter, weigh up to 1,400 pounds and require cranes to lift?

The Bell system faced the problem of trying to get the correct assorted-size wire and cable reels to show up at the right factory to be reloaded and sent back out again to the phone companies.

Besides obvious transportation costs of shipping large reels from one manufacturing location to another to correct shortages and surpluses, Bell incurred additional costs for stored excesses and disrupted manufacturing schedules. For help, Bell called a computer.

By using mathematical models for mapping fluctuations in reel shipments and transportation charges for six different types of reels, the computer anticipated changes in supply and demand for each type of empty reel.

The system includes 36 Western Electric service center and telephone company areas, which receive shipments of full reels, and five manufacturing locations, which produce the cable or wire. With the computer's forecast of supply and demand values, the computer creates a shipping schedule for empties to meet each factory's needs at the least possible shipping costs.

Bell's application of computer technology to reel traffic has proven effective — lessening managerial headaches, cutting company expenses, and ultimately, saving consumer dollars. Interplant shipments decreased — to the tune of about $1 million in savings a year.

In other words, their computer system works reel good.

Look Ma, no cavities

When did it all begin? With cavemen who used fish bones to floss? With ancient sultans who permitted long-nailed concubines to extract food particles from between their royal molars? No one really knows for sure, but society has been paying duty to its pearly whites for a long, long time. Now what has modern technology done? They have advanced one step further — to a computer-controlled toothbrush.

The idea for this "Intelligent Toothbrush" belongs to Bill Newcomer, Dylakor Software Systems president. And if you're willing to "invest several thousand hours of spare time" and "under $10,000 at your local computer store" he'll gladly supply you with the details for assembling your very own Zilog Z80 microprocessor-controlled intelligent toothbrush to take care of the intricate back and forth, inside and outside, up and down, forward convolutions, double helices and rapid random rotations involved in brushing.

On the hardware requirement list Newcomer mentions these items: one IBM 370/168 computer, or the equivalent; a tube of Pepsodent toothpaste (mint flavored works best), and a fire extinguisher. Software requirements include a pencil, one large eraser, a hexadecimal to metric converter and three or more used memory prints.

The do-it-yourself plans for building your toothbrush come with a brief five page, single spaced instruction booklet. The booklet includes a section on "How to relate to you smart toothbrush" (since it may be smarter than you), and a single (rejected) microprocessor chip that becomes the "cheapest part of the microcomputer".

The glossary supplies definitions to such commonly used

computer terms as: "RS232 — the 25 pin socket that lets you plug a $3000 video terminal into a $10 microprocessor".

Copies of the booklet are available while they last. So if you're interested in learning how to take care of your Pepsodent smile via modern technology, write to: Dylakor Software Systems, Inc., 26255 Ventura Blvd., Encino, CA 91436.

Complying with Consumer Complaints

Today's marketplace won't tolerate poorly supported home computing systems very long. Home computer suppliers (manufacturers and retail outlets) will have no choice but to provide "some reasonable type of support" and fully debugged software, according to Jon A. Turner, director of advanced systems at the Center for Computing Activities, Columbia University, New York.

A teacher and researcher of information systems at the university, Turner also served on the Office of Management and Budget task force that studied and recommended organizational changes to the office of the President.

Turner believes computer "literates" (his term for computer technology-oriented people) will provide the initial market demand for home computer systems and absorb what's available on the market. Computer "illiterates" (people with little technical background), he continued, will wait until consumer appliance-type warranties and reliability are guaranteed. Turner also believes that companies should not rush into production and marketing with machines that don't work properly in the first place.

Many businesses accepted untested and unreliable computers because the first mainframes were "oversold" by manufacturers. Turner commented, "The projected benefits were so great in terms of labor replacement, efficiency and general economics that business, unwisely, was willing to tolerate the condition and thus a precedent was established."

If businesses had challenged manufacturers in the beginning, we might not be in our present situation, Turner speculated. But, he said, you could argue that an early challenge might have slowed technological development.

As the personal computer becomes more popular for home and small business use, Turner foresees the need for a variety of maintenance arrangements. "I'm not sure that anything substantially less reliable than your television set is going to be tolerated."

But Turner doesn't foresee personal computers having the same impact as television. Television probably would not have had the impact it did if radio had not preceded it.

"After all," he continued, "it took a long time for the telephone to really change patterns of movement within communities." Turner suspects "several decades passed before the pattern of visits and dropping-in on people were replaced by telephone calls.

Home computers may have changed lifestyles to some extent, but Turner doesn't think the overall future effect will be significant. "People will continue doing what they're doing. It may be that some people, on the more creative end, will find some applications which, over a long period of time, alter what people do, but I don't see it coming about quickly."

Man and the Sea

Inspection of underwater pipes and oil drilling platforms, and searching for and retrieving objects from the ocean floor present many problems in terms of divers and manned vehicles. The main problem, of course, is man. With a limited workday, high costs and great dangers, even elaborate life-support systems don't fill the deficiency.

But Hydro Products has the next best answer short of a porpoise with an engineering degree — a highly "intelligent" microcomputer-controlled undersea robot vehicle that has eyes, ears and arms with flexibility in many applications. Called the RCV-150, it is one of the more advanced commercially-produced submersibles in use today.

Two identical PACE microcomputers, one in the undersea vehicle and the other on the surface mother ship, manage virtually all communications, display and control functions.

The undersea portion of the system consists of a small cable-tethered vehicle propelled by four thrusters. The vehicle's small size makes it easy to launch and retrieve.

With only 50% of the available space in use, options such as marine magnetometers, acoustic trackers, sonar depth-finders, 35mm still cameras, search sonars or even a second manipulator arm can still be added.

"Our software and firmware allow us to add on any number of features to meet the specific requirements of any customer application," explained Jim Tierney, a Hydro Products senior EE. "Control firmware can also be supplied on programmed read-only memory (PROM) for convenient field installation."

Internal depth and yaw programs referenced to depth and turn-rate signals are also microcomputer controlled to relieve the operator of many maneuvering tasks, allowing him to concentrate on observation, inspection and manipulation duties.

Vehicle development took nine months; about 1.5 man-years went into software development.

Three PACE board-level computer cards comprise the vehicle's microcomputer while the same three plus a second PROM board make up the surface control station microcomputer.

PROM memory firmware houses operating programs for vehicle and control units. The second PROM board in the control station contains maintenance and self-test programs and CRT display formats. RAM boards temporarily store intermediate results during processing.

The surface-ship control station has two monitors. One shows video pictures from the undersea television camera, while the other displays operations data.

Illuminated bar graph displays in the control station indicate power levels and temperatures. Illuminated switches indicate control position and serve as warning annunciators. Continual monitoring of vehicle performance by the microcomputer immediately reports problems to the operator. Data collected by vehicle sensors and detectors is transferred through the microprocessors for surface display.

The development system will be used for future program changes and as a stand-alone computer performing tasks such as maintaining parts list, running management programs and making engineering computations.

Not exactly a mammal with a PhD, but it does the job.

Educational Applications

Computer applications in undergraduate college and university education will be the topic of a national conference at the University of Denver June 12-14.

Agriculture, business, history, humanities, languages, mathematics, education, engineering, fine arts, psychology and statistics are just a few of the scheduled conference discussion topics.

Actual experiences using computers in these and other subject areas are being solicited.

For more information write to: Dr. William Dorn, Department of Mathematics, University of Denver, Denver, CO 80208.

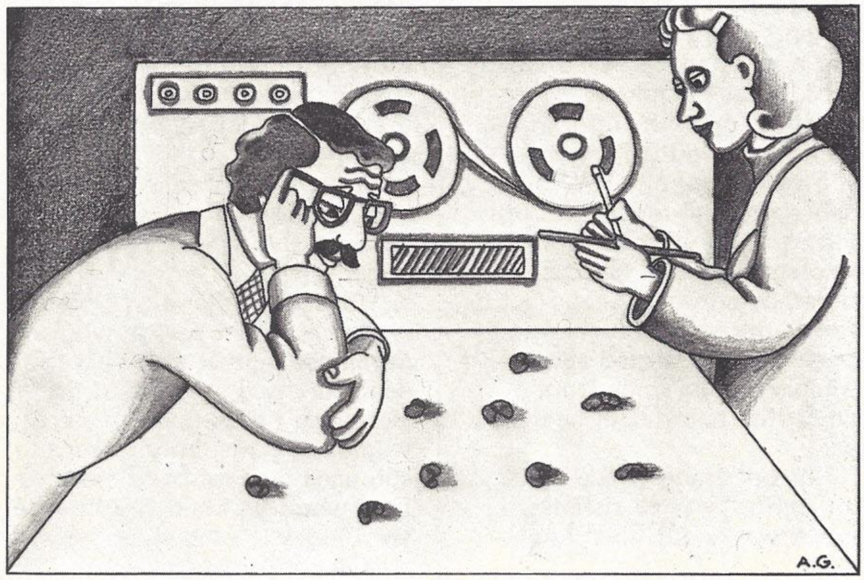
Groundhogs, Bunions, and the Woolly Worm?

Want to know how cold it will be next winter? Don't worry about checking the *Farmers' Almanac*; study the woolly worm!

That's what a team of biology students at Appalachian State University (ASU) in Boone, NC, are doing to predict the severity of the cold weather season.

For the past two years, the University's Center for Woolly Worm Studies has been collecting the furry insects (which are actually the caterpillar stage of the Tiger moth) and counting the black and brown colored bands on their bodies. According to folklore, coloring on the bands is an indication of weather. The more brown bands there are on the insects, the milder the weather.

Data collected this year on the woolly worms' coloring pattern, together with weather information, will be fed into the University's Sperry UNIVAC 90/60 computer system to determine if temperature, humidity and rainfall — believed to affect the insect's coloring — also have an effect on the coming winter.



The Center's findings concerning the winter of 1976 were accurate but only partially correct for the 1977 winter which turned out to be one of the coldest ever on the East Coast.

According to Dr. Sandra Glover, head of the center and an ASU biology faculty member, about 50% of the worms collected last year showed mixed bristles on bands with both black and red coloring.

"We speculated that this could mean extremely erratic weather. It sure did!" noted Dr. Glover.

Interest in the center's research resulted in people collecting woolly worms throughout the Eastern states and shipping the insects to the biology department at ASU.

Teaching "Toys" for Tots

Back in November, 1975, Joseph Willhide, a 37-year-old associate professor of systems and computer engineering at Boston University's College of Engineering, was sitting in his house in Yarmouth, Cape Cod, when he was struck by an idea:

"Why not program a small computer to be a toy for children?"

Willhide grabbed a scrap of paper, scribbled some rough designs, and called his wife Evvie over to have a look. She liked it, and Willhide went to work.

This past Christmas the whole country got a chance to look. That's when Willhide's finished product, "The Mathemagician", went on sale through Sears Roebuck, J.C. Penney's, Montgomery Ward and Dixon's of England, among others.

The Mathemagician is a combination of a small computer and colorful plastic overlays with titles like Lunar Lander, Gooey Gumdrop and Walk the Plank that transform arithmetic into games, contests and feats of derring-do. Willhide believes children will soon play with computers with the same excitement other generations once played with model railroads.

Computer play has tremendous educational benefits, Willhide says. "Relying on a calculator often retards a child's mathematical development," he says, "but playing with The Mathemagician accelerates it. A bright child can literally race through several years of arithmetic with it."

The basic component of Willhide's "toy" is Texas Instruments' TMS-1000, a microcomputer the size of a stick of gum. The toy has roughly the same capabilities as the original electronic computer, ENIAC, which weighed 30 tons, was housed in a room 50-feet square at the University of Pennsylvania and dimmed every light in West Philadelphia whenever it was turned on.

"As recently as World War II nations would have paid millions for a computer with these capabilities," Willhide says. "Today the basic components sell for under $3."

The route from Willhide's bright idea on Cape Cod to the sales catalogues of the world's biggest merchandisers has been, in the Horatio Alger tradition, "a long, hard road." Willhide, Evvie and a partner, Henry Viarengo, worked for months to create a working model. Comput-

er innards covered the dining room table for weeks while the three ate off TV trays.

"Henry and I worked on the thing so constantly," Willhide laughs, "that the joke was that he'd throw open the door in the evening and shout to my wife, 'Hi, honey, I'm home!' "

Once the first model was finished, Willhide started knocking on executive doors. He was scorned by admitted manufacturers of "junk toys", laughed at by advertisers, and turned away with a condescending smile from many office suites. But he didn't quit.

"The one thing I had," he says with a grin, "was persistence. I **knew** it was good! I just **knew** it!"

At last Willhide stumbled onto APF Electronics, a calculator company in New York City. APF decided to gamble. They gave Willhide a contract and unveiled The Mathemagician on June 5, 1977, at the Summer Consumer Electronics Show in Chicago, an event which draws about 40,000 dealers. The Mathemagician promptly walked off with the show's Design and Engineering Award, and APF is convinced that at $39.95 they've got a winner.

In the meantime, Willhide is hard at work on still another toy idea, which he prefers to keep under wraps.

"Basically," he says, "I'm an under-confident person who's developed a great deal of confidence in my own sense of timing. Years ago I developed an electronic TV game, only to be discouraged by a lot of toy executives who told me it would never sell. Today it's the biggest growth area in consumer electronics, targeted to be a billion dollar industry."

After The Mathemagician, it's unlikely that anyone will ever again be able to dissuade Joe Willhide from pursuing a good idea.
— Don Clark, Bostonia *magazine*

He has a mind like a computer . . . makes errors faster than anyone else at the executive level.

Let your computer do the thinking . . .

For the personal computerist on a low budget, there's a new third generation (no programming skills necessary to implement tasks) computer with a preprogrammed library of educational, home management and entertainment programs — VideoBrain.

The computer comes equipped with AC adapter, TV hookup cord, antenna switch box, two joy sticks and three introductory cartridge programs. You can hook it right up to your TV and start running programs. The console has 36 input keys, weighs 10 lbs., and uses about as much energy as a clock radio.

Available through department stores and specialty electronics shops, the FCC-approved computer will sell for $500 (suggested retail price).

Basic text and timekeeping programs are built into the computer. The text program allows the user to type and edit 7 line, 16 characters per line messages, change the color of the screen or the size of the letters and store messages for retrieval. The calendar program keeps track of the year, month, date, day of week, hours, minutes and seconds, while utilizing other programs.

The 8-bit microcomputer contains 1K bytes RAM and 4K bytes ROM, while the preprogrammed cartridges can hold up to 13K bytes RAM or ROM.

Your TV set serves as the computer's output, taking advantage of the TV's color range and sound capabilities. LSI circuits allow high resolution displays and tonal quality.

Twelve easy-to-use, ready-for-market programs include: "Finance I", a program for analyses of loans, mortgages, savings accounts and other financial alternatives; "Cash Management", a record and summarization of monthly household and income spending, credit cards and charge account standings; "Stock Valuation I", which evaluates stocks

on the basis of projected future dividends, and allows users to compare stocks based on multiple assumptions of growth, probability, shares outstanding, and dividend policy; plus Gladiator, a challenge game with 384 variations; Blackjack; Checkers; Pinball; and more.

According to Dr. Albert Yu, 50 more programs are in the developmental stage and will be introduced at a rate of three or four a month during 1978.

So why not indulge yourself and let someone else do the thinking for a change?

Machine Trust

Everyone has his favorite story about what will happen if we put too much trust in machines. But now there's a collection of 16 short stories compiled by D. Van Tassel on the subject. The book, *Computers, Computers, Computers: In Fiction and in Verse* includes works by Art Buchwald, Robert Heinlein, Bob Elliot, and Ray Goulding.

One story, "2066: Election Day", tells you about the computer that's responsible for testing anyone who wants to be president of the United States. There's one catch though. Since the President's job has become so complex, the computer decides to make the test impossible to pass.

Then there's a computer-error-sends-innocent-man-to-jail-and-another-computer-prevents-his-release story by Gordon Dickson.

Robert Sheckley's story, "Fool's Mate", describes a computer so concerned with looking for the perfect battle plan that it overlooks the obvious.

All the selections are humorously believable — even to people familar with computers. So for people working with computers or for those just entering the field, this book promises to be fun. Published by Thomas Nelson, Inc., 30 East 42nd St., New York, NY 10017 192 pages.

Patient's Liberation

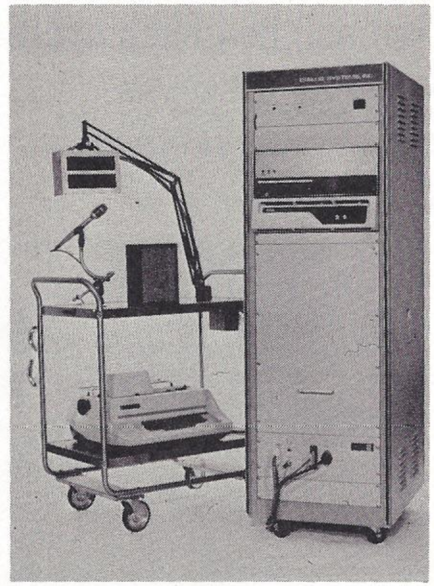
Being laid up in the hospital used to mean total dependence on family, friends and hospital staff. But patient's lib has arrived at a number of hospitals.

Through a new voice control system, the Dialog 117 System by Dialog Systems, Inc., immobilized patients can control bed motors, lights, typewriter, telephone, calculator, computer games, television, radio, and nurse call by speaking commands into a microphone.

With a vocabulary of 99 words, the system verifies commands, prompting, and other communication with the user through an electronic display. The system converts sound waves into electrical signals which in turn are amplified and converted into a digital form that can be understood by the system's word processor.

Because the system trains itself to the user's manner of speaking, it can respond to foreign accents and dialects as well as understand some patients' words which attending nurses might not comprehend.

One of the more interesting capabilities of the system involves the typewriter, which will



type a personal letter as the user spells each word.

The system gives paralyzed patients' increasing independence and opportunities for self-controlled occupational therapy. It could also help keep hospital costs down by freeing the services of nurses and therapists.

Although the 117 System services only one patient, custom ized multi-channel systems (capable of handling the needs of eight patients) can be built by special order. For more information, contact Dialog Systems, Inc., 32 Locust St., Belmont, MA 02178.

Wall Street Wizardry

Wall Street can get crowded at times and access isn't all that easy. General Telephone & Electronics Corporation has introduced a computer-controlled information and communications system designed to improve the productivity of the brokerage industry.

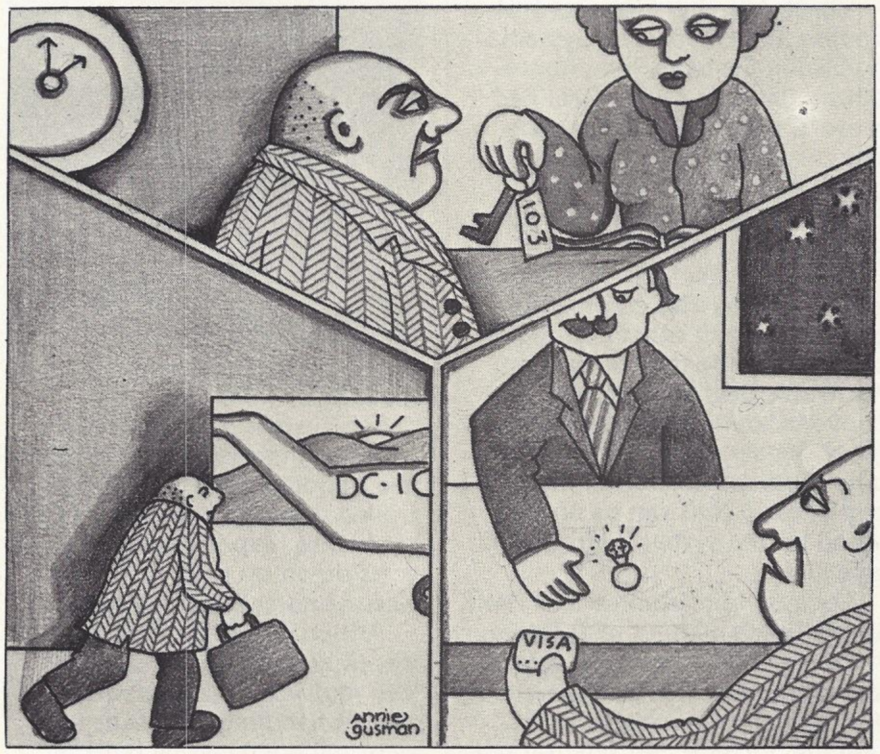
Called the GTE Financial System One, the machine brings all relevant information right to the broker's desk through a video-display terminal. In addition, the system permits instant communications, via the terminal, between headquarters, branch and satellite offices.

Connected to a customer's private network, account information can be read at authorized desk terminals. The brokerage firm controls which units have access to the private data base.

In its information mode, the system provides quotations on 16,000 securities, bonds and commodities in a number of different formats. It also provides instant access to, and daily updates on, Standard and Poor's Stock Guide Retrieval Service which provides statistical and business data on 5,200 securities; newest research reports on securities, bonds, money markets and options; and security positions and market activities of major banks and investment companies.

And the system doesn't make nasty remarks if you remove your tie at your desk . . .

'round the clock credit check



Checking into a hotel late at night, taking an early morning flight and making an evening purchase at a shopping center three time zones distant all make credit card authorization an around-the-clock business.

For the Eastern States Bank-Card Association (ESBA), this requires a reliable and available data processing system — a minicomputer (the Tandem NonStop 16 dual-processor system) to field the constant flow of inquiries.

As front end to an IBM/370-158 mainframe, the dual-processor routes incoming requests for credit on Master Charge and VISA cards directly to the computer for processing.

Should the ESBA system or host communications lines fail, the Tandem can handle all credit authorization inquiries by itself. In this event, the system refers to its disk files for delinquent accounts and other "bad guys" listings, which are updated daily with magnetic tapes.

If a specific processor (there can be up to 16) should malfunction, another instantly assumes its duties without operator intervention — hence the name, NonStop. Repairs can be made with the system operating.

Two 50-megabyte disks and internal memory of 256K bytes each comprise the dual-processor system. The two systems are usually arranged one on line and one off line, although they can be used to back up each other. One single data file accessible by both systems contains some 2.3 million records stored on disk.

When member merchants telephone the association they tell terminal operators their account number, which is keyed into one of the 80 connected terminals. The computer then routes communications to the appropriate mainframe by referring to a table in memory based on account numbers. If the number corresponds, the credit inquiry is sent to the local system. If the line is not available, the system refers to its negative file disk for a quick determination, reporting back through terminal operator's CRT screen.

So if you were considering pulling a credit card con by skipping out on your bill at the Honolulu Hilton and catching a plane to Argentina, forget it.

Look It Up

Everyone's read an article or seen a program they'd like to remember — only to forget the next day where they saw it.

To help solve this problem, you can now order a copy of the *Periodical Guide for Computerists.*

The Guide indexes articles, letters from readers, book reviews and editorials relevant to the personal computing field from at least 20 popular computer-oriented magazines including *Personal Computing.* It is available as a complete January-December 1977 index for $5 or January-June and July-December for $3.50 each. A 1975-76 index of magazine articles is also in the works.

To place an order (COD or cash with order), write to E. Berg Publications, 1360 SW 199th Ct., Aloha, OR 97005.

Come out of the Closet

Show us and our readers what your home or small business computer setup looks like. Whether you've housed your micro between the filing cabinet and boxes, on top of the refrigerator or in a custom designed cabinet.

We're also interested in how you, your company or someone you know may have used computers to help in unusual applications. Or, perhaps you have a humorous story to tell. We'll follow up on your leads, verify them and print the story for everyone to read about.

So send your photos, stories, or leads to Random Access, Personal Computing, 1050 Commonwelath Ave., Boston, MA 02215.

This 8-bit machine, by itself, is as versatile as a lot of systems that include peripherals

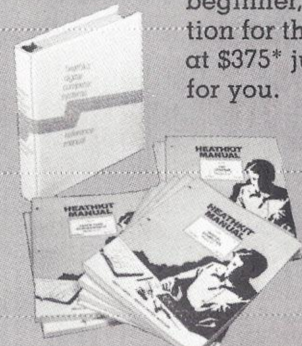
Skeptical? For starters, because of its unique design, the H8 is the only machine in its price class that offers full system integration, yet, with just 4K of memory and using only its "intelligent" front panel for I/O, may be operated completely without peripherals!

Memory Display

Register Display

I/O Port Display

In addition, by using the features of its built-in PAM-8 ROM panel control program, the H8 actually allows you to dig in and examine machine level circuitry. Responding to simple instructions, the "intelligent" panel displays memory and register contents and lets you inspect and alter them even during operation. And for greater understanding, the front panel permits you to execute programs a single instruction at a time. The H8's memory is fully expandable, its 8080A CPU extremely versatile, and with the addition of high speed serial and parallel interfacing you gain the added flexibility of I/O operation with tape, CRT consoles, paper tape reader/punches, and soon floppy disk systems! The H8 offers superior documentation including complete step-by-step assembly and operation manuals, and comes complete with BASIC, assembler, editor, and debug software that others charge over $60 for! H8, simplicity for the beginner, sophistication for the expert and at $375* just right for you.

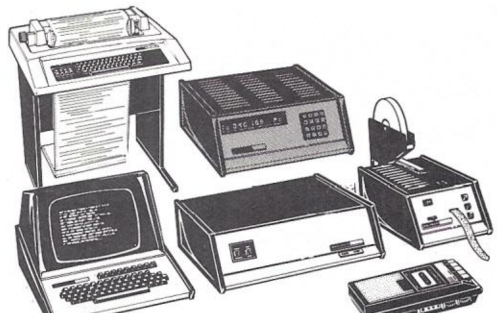


*Prices are mail order net FOB, Benton Harbor, Michigan. Prices and specifications subject to change without notice.

HEATHKIT COMPUTERS

System Engineered for Personal Computing

H8



Computers, peripherals and nearly 400 exciting, easy-to-build electronic kits, all in your

FREE Heathkit Catalog

HEATH Schlumberger

Heath Company, Dept. 352-390
Benton Harbor, MI 49022

Please send me my FREE Heathkit Catalog.
I am not on your mailing list.

Name________________

Address________________

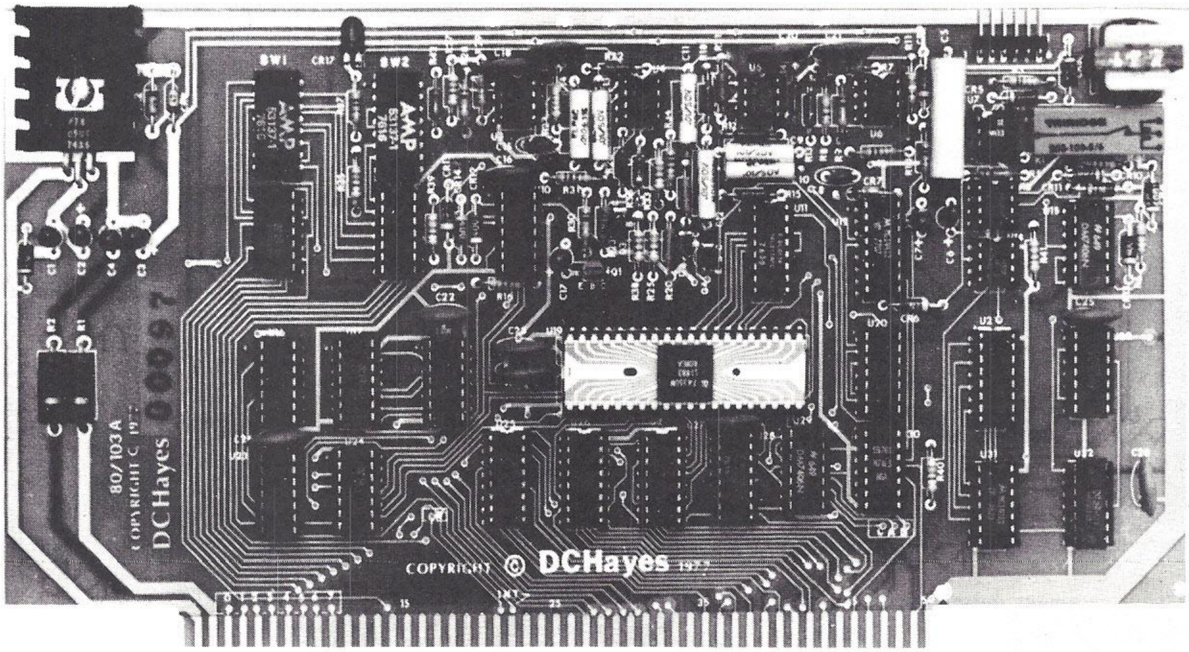
City________________ State________

CP-143 Zip________

TIMESHARING

The 80-103A works both ways. Your system can call a timesharing service and communicate as an intelligent terminal *OR* your S-100 system can be the timesharing system where the 80-103A answers the phone and communicates with terminals or other processors.

80-103A DATA COMMUNICATIONS ADAPTER



The 80-103A DATA COMMUNICATIONS ADAPTER was developed to function as an S-100 bus compatible serial interface incorporating a fully programmable modem and Telco interface. These functions are usually accomplished by the use of two separate modules: 1) a serial I/O board, and 2) an external modem. By combining these features on a single board, the 80-103A can offer microcomputer applications significant cost/performance advantages over other implementations.

- FULLY PROGRAMMABLE FEATURES
- AUTOMATED DIALING AND ANSWER
- ORIGINATE OR ANSWER MODE
- 110-300 BIT/SEC DATA RATES
- CHARACTER FORMAT AND PARITY
- ERROR DETECTION
- FULLY BUFFERED, OUTPUTS DRIVE 25 S-100 BUS LOADS
- STANDARD U.S. FREQUENCIES
- FULL TELCO COMPATIBILITY WHEN ATTACHED TO DAA
- COMPATIBLE WITH EXISTING TELETYPES AND TIME SHARING MODEMS
- ALL DIGITAL MODULATION AND DEMODULATION. NO ADJUSTMENTS REQUIRED.

DC Hayes Associates offers a full range of capabilities for solving your information handling problems. Whether your problem is large or small, we will apply innovative techniques for finding the best solution. Contact us about our products and services.

PRICES:
Assembled 80-103A with 48 hour burn in and 90 day warranty is $279.95

Bare Board with manual is $49.95

Dealer Inquiries Invited

H20

D C Hayes associates

P.O. Box 9884 • Atlanta, Georgia 30319 • (404) 231-0574

CIRCLE 10

Testing ESP by Computer

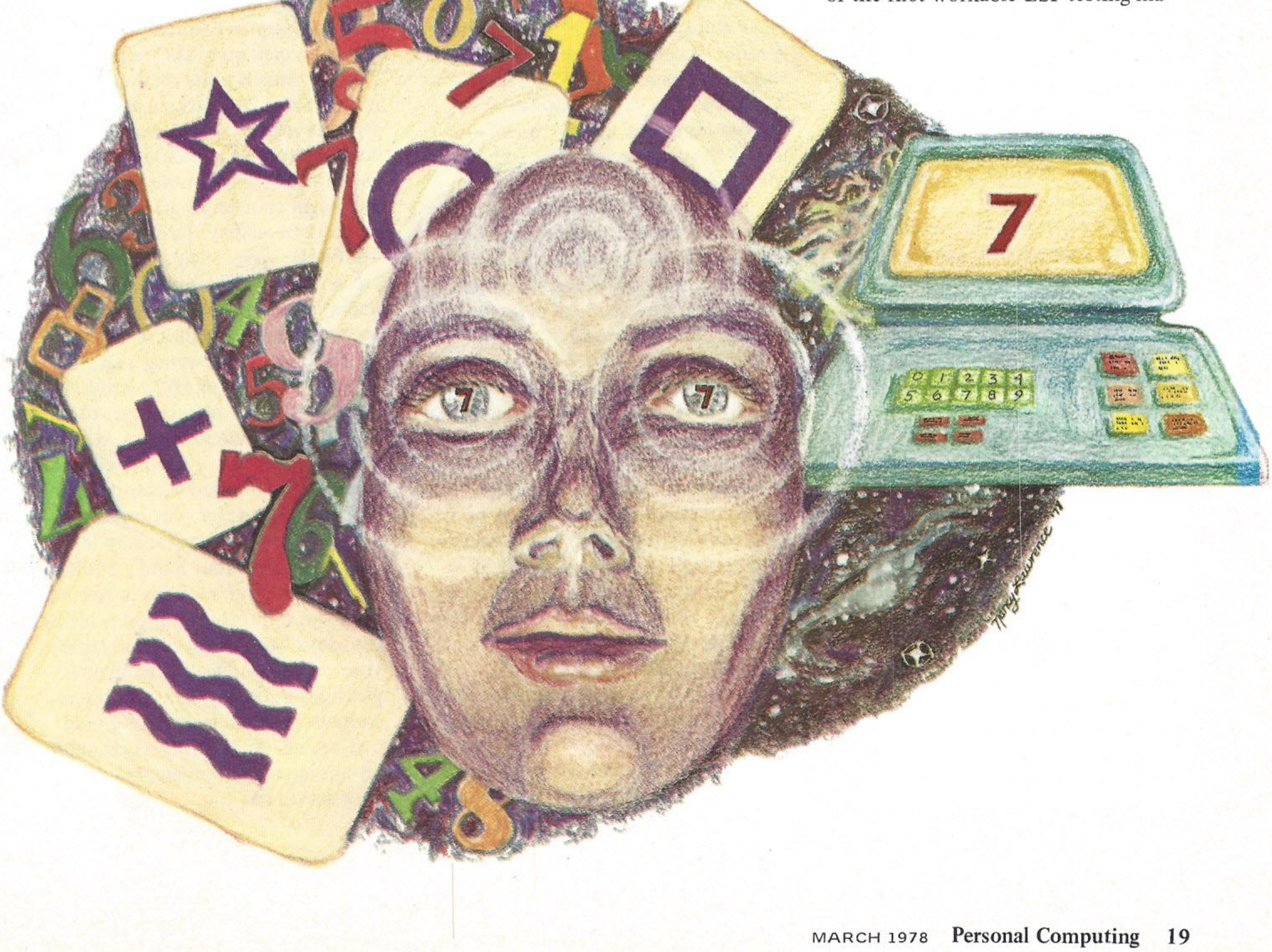
BY DAVID GALEF

Personal computing continues to expand as new uses for computers emerge. Programs have been written to deal with problems from accounting to zoology and beyond. The randomization effect possible on any home computer system allows for the design of everything from games to experiments. One particularly intriguing application involves a device for detecting extrasensory perception (ESP).

Most early testing procedures in ESP were slip-shod, allowing for a number of errors during examination; true randomization of numbers and card selection was not achieved, facial expressions of examiners often hinted at the desired outcome and recorders frequently made unconscious errors in transcribing the results. The answer to all these problems, of course, was simple – computers. Long used in other fields of science, computers could be applied to parapsychology.

Since the early days of parapsychology research many prototype machines have emerged: the VERITAC, Aquarius, and ESP-1. Systems with features similar to these machines can be incorporated into any home computing set-up.

In 1962, a technician at the Air Force Cambridge Research laboratories built the system called VERITAC, one of the first workable ESP testing ma-



Since the early days of parapsychology research many prototype machines have emerged: the VERITAC, Aquarius, and ESP-1. Systems with features similar to these machines can be incorporated into any home computing setup.

chines. The set-up was no more than a sophisticated random number generator, with a few adaptions made to fit ESP testing procedures. To test precognition – seeing events before they've actually occurred – the VERITAC would select a number from 0 to 9 after the subject had made his guess. To test clairvoyance, the machine would choose a digit before the subject's guess to see whether, under the definition of clairvoyance, the subject had empirical knowledge of events that had already happened. For telepathy testing, a person remained in a sealed room with VERITAC and attempted to "send" a number choosen by the computer to a subject in the other room. In this particular phase, however, clairvoyance (just "knowing" the number) played a part.

VERITAC scored the results automatically, removing human bias. In the end, ESP scores were actually lower when VERITAC was used, a phenomenon many parapsychologists blamed on the machine's "inhumanity", which they claim discourages ESP talent.

Experimenters continued to use VERITAC despite its disappointing results since it was the most controlled operation they had. For each trial the printout displayed the selected number, the subject's guess and the time the subject took to make this estimation – all impartially, quietly, and without fuss.

A later model of a similar type computer system was developed by Russell Targ, an electronics technician, in conjunction with David B. Hurt. The machine, Aquarius, operated on a strictly numerical system, like VERITAC, using random numbers from 0 to 9. But Aquarius differed from its predecessors in its responses to subject performance. In the past, many parapsychologists had complained of machine "unresponsiveness". The Aquarius system countered these complaints by encouraging high performance with reinforcement.

Five lights were used to spur the subject on to make more correct guesses. The light "Good beginning" flashed with six correct guesses, and "ESP ability present" lit up at eight. A series of similar encouragements was given along the way: ten hits – "Useful at Las Vegas", twelve – "outstanding ESP ability" and fourteen – "Psychic, medium, oracle". Also like the VERITAC machine, Aquarius printed its results on a roll of paper in a presumably tamper-proof method. By setting the machine in the proper mode, the experimenter could test for evidence of clairvoyance, precognition or telepathy.

The last machine, recently reported in *Parapsychology Review,* is a model called ESP-1, made by Paratronics, Inc. ESP-1 has a few extra features as well as being somewhat more portable than earlier models. While it lacks Aquarius's system of flashing lights, it does have two feedback monitors, one of which automatically gives the number of "hits" (correctly guessed numbers) per ten trials, while the other informs you of the hit score any time you press a certain button.

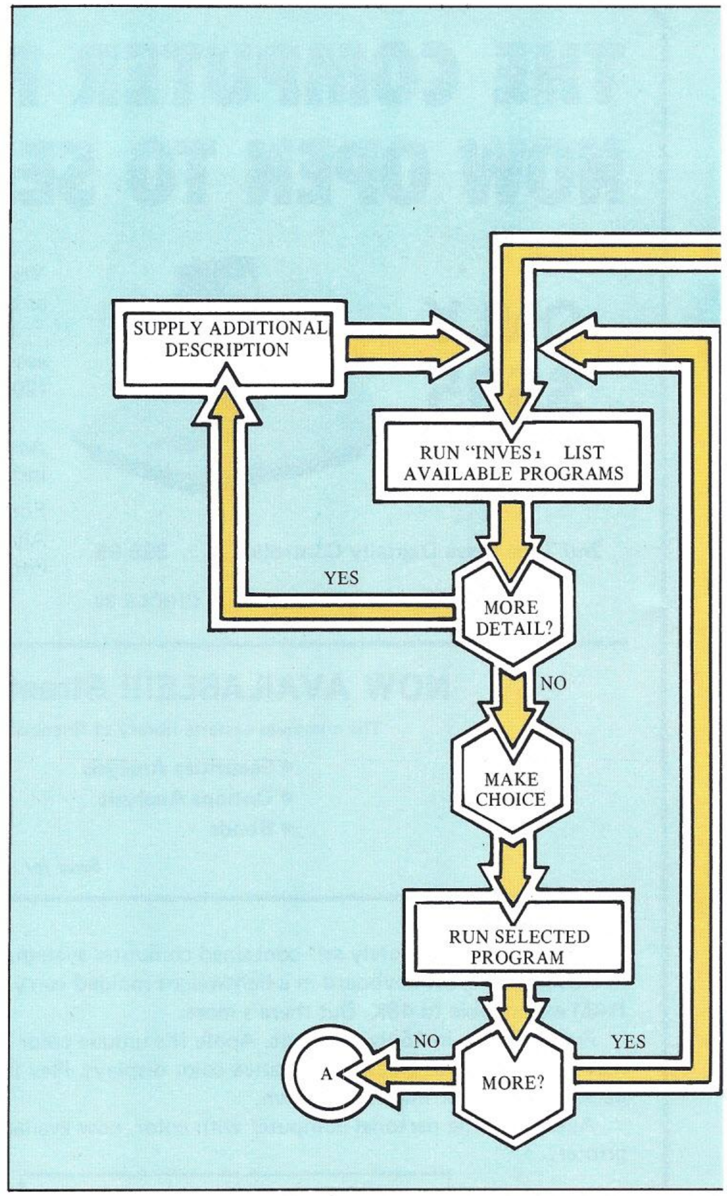
The subject chooses a number from 1 to 4, so that a chance average would be 2.5 hits out of 10 guesses. The ESP-1's most distinguishing characteristic, a series of two random oscillators, mixes the numbers to be picked at an indeterminable rate. As long as the subject holds down the activating button, the numbers continue to oscillate in random array. When the subject releases the button the most recent choice automatically appears. But this feature also introduces a new type of human control: the subject can try to influence what number the machine will pick by holding down the button for a certain length of time.

Unfortunately, all these elegant controls are inelegantly biased. The ESP-1 apparatus seems to have a tendency to pick numbers 1 and 3 over 2 and 4. Possibly the oscillators do not perform their job correctly. Whatever the reason, during a series of tests the subject will either consciously or unconsciously note the aberration and begin choosing 1 and 3 with increasing frequency. Intentionally or otherwise, biases such as these tend to creep into a field where scientists and manufacturers alike are fervently hoping to produce evidence of ESP.

The home programmer can adopt his own methods of ESP testing. With a suitable printout technique, the five Rhine cards – circle, square, wavy lines, star, and plus sign – can be replicated. You can program your computer to pick out the integer portion of a random number. The easiest way is to input numbers from Rand's book of one million random digits in five-number groups, but this method is rather laborious. And besides it will stifle your creativity. One of the main pleasures in testing ESP on a home computer system is designing your own set-up. And if the real computer set-ups allow for human error and bias, your set-up can too.

Program Linking

BY O.E. DIAL



A programmer's appetite for computer core memory is insatiable – particularly in a microcomputer environment. For most builders, the microcomputer develops incrementally. First, the essentials with minimal memory boards. Then, as money becomes available, additional 4K memory board kits. There comes a time, however, when the builder faces a choice – additional memory boards or a disk drive?

This choice occurs when the builder has about 28K to 32K of memory. Of this, about 20K will be consumed by the interpreter, reducing available core memory to 8K or so. At this point, if you do choose to buy a floppy disk system, you will add 330K addressable bytes to your memory storage.

This unfortunate loss of memory parallels a trend which makes it even worse. The micro-bug tends to deal with progressively longer and more complex programs, reflecting an advanced understanding of problems and programming techniques. The programmer will continue to challenge the limits of the machine's capacity and return again and again to the requirement for the 4K boards, possibly exhausting the available slots on the mother board before reaching the 60K limit (assuming that the top 4K are reserved for ROM, e.g., a bootstrap loader).

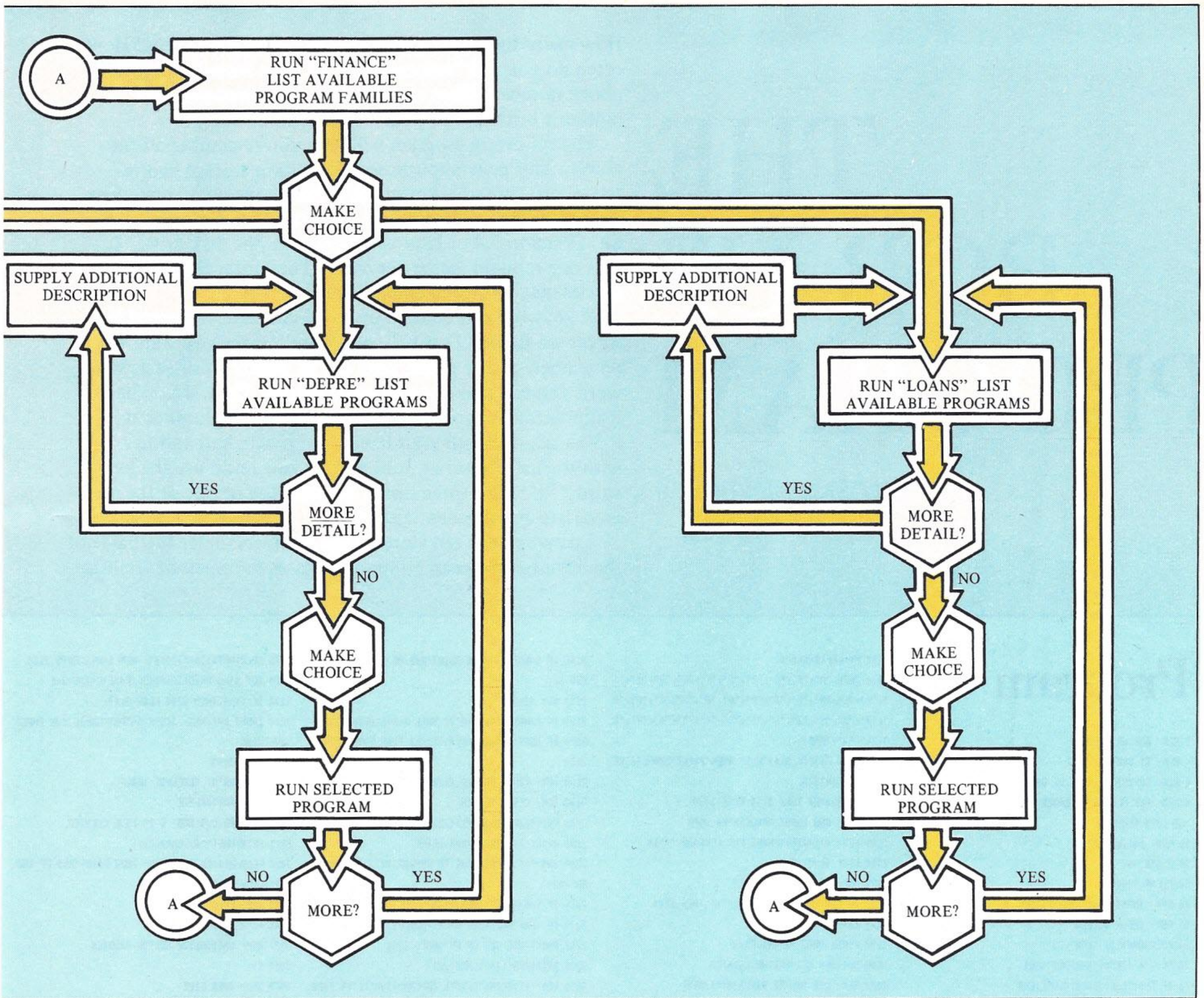
But you can reduce core requirements even when dealing with very long programs by using a method called, for want of a better name, program linking. The technique is simple. It consists of dividing a program into a number of disk-stored programs and letting one program call the other when its turn comes until all program segments have been run. The programmer must be careful to file data generated in one segment but needed in another. The same file may serve all program segments.

The technique relies on only one simple programming instruction – "RUN name of program segment". Assume, for example, a program divided into ten segments. The last instruction in the first segment is "RUN SEG-2". The SEG-2 program, similarly, concludes with "RUN SEG-3". This process continues until all segments have run.

Grouping programs relating to the same application area into families from which the user may make a choice of the one needed at a particular time is a natural application of this technique. For example, a great many discrete problem-solving programs exist in the general area of finance. Some programs relate to investments, other to loans or depreciation calculations. The user may group those together in a pyramid of programs for convenience.

Take, for example, the programs contained in *Common Basic Programs* by Lon Poole and Mary Borchers (Osborne & Associates, Inc.). This book provides programs most commonly used in investments, loans, depreciation schedules, statistics and math, together with a few miscellaneous others. The language is BASIC, but because of differences in the powers of BASIC interpreters, the authors programmed at the lowest level of commonality, thus minimizing the need for editing. For example, they avoided SWAP and PRINT USING statements, among others.

The book contains some 19 financial programs. Generally short, the programs (15 to 30 statements) require very little core. Nine programs deal with investment problems



(the future value of an investment); three deal with depreciation (salvage value); and eight deal with loan problems (annual interest rate on a loan). Use these programs as a foundation and build two program levels on top. At the top level a very short program simply asks the user to indicate interest in one of the three areas – investments, depreciation schedules or loans. We can call this program "FINANCE". The user makes a choice and FINANCE calls for the selected program to run.

At the mid-level are three programs: "INVEST", "DEPRE" and "LOANS". These programs serve as a directory to the programs embraced by each. Further, they describe each program's use. INVEST, for example, displays the full titles of its nine programs, and asks the user whether he desires more information about any of the programs. The user selects the program he wants to know more about.

The computer provides a paragraph-length program description, then repeats the query. If the user declines more information, the computer asks the user to select the program to be run.

To this point, the computer has linked from a program at the top of the pyramid to one of three mid-level programs. By making the last selection, the mid-level program links to a base-level program with a RUN statement. When the base-level program has run, the computer asks the user if he has other problems requiring this program. If not, the base-level program links back to the program at the top of the pyramid by a "RUN FINANCE" instruction. Returned to the top of the pyramid, the user may again select from the three families of finance programs.

Taken altogether, this technique requires some 500 program statements at the base level. Yet the computer never sees a program longer than about 80 statements. The mid-level (which you can think of as overhead) requires about 10 statements plus strings. These strings can be burdensome if you include lengthy descriptions of each base-level program. Brief paragraph headings of each program from Poole and Borchers' book seem to suffice. The top-level program requires only about 10 statements.

After exercising the technique, you will recognize numerous break-points in most long programs. These breaks indicate natural candidates for program segmentation and the use of RUN instructions for program linking. Practicing the technique on long programs can mean the difference between a good run on one hand, and receiving the egregious OUT OF MEMORY advisory on the other.

THE BOB-UP PROGRAM

BY SAM NEWHOUSE

How many times have you forgotten an appointment? How often do you call "information" to get a friend's forgotten phone number? What about locker combinations, your mother's birthday, or your wedding anniversary?

The following program will help you remember all the above. This general purpose information storage and retrieval program uses keywords as handy memory references. As written, the program runs on Altair Disk Extended BASIC (Version 3.4 or later versions) with one disk drive. It uses one random sector per entry. Therefore, about 2000 entries may be stored and indexed.

A keyword can contain up to 15 characters. Typical keywords would be "Dentist", or "Mom", dates you want to remember, or any name, such as "M. Snerd". For each keyword you can make one or more entries. Thus, if you had 3 appointments on 5/15/78 – one with the plumber at 9:30 a.m., one with your lawyer for lunch, and another with the hairdresser at 3:30 p.m. – you could use the keyword "5/15/78" three times, entering the details of the appropriate appointment each time.

Likewise, you can store phone numbers under the name of the business or person; important dates to remember remain

Program

```
2 REM- BOB UP PROGRAM
4 REM- BY SAM NEWHOUSE
6 REM- COPYRIGHT 1978 BY SAM NEWHOUSE
8 REM- FOR ALTAIR EXTENDED DISK BASIC, ONE DISK,
 CT-1024 TERMINAL
10 REM- INITIALIZE
20 WIDTH 48
30 CLEAR 5000
40 REM- INPUT COMMAND ROUTINE
50 REM- CLEAR SCREEN
60 PRINTCHR$(16);CHR$(22);
70 CC$="*":INPUT"COMMAND";CC$
75 IF CC$="END" THEN CLOSE:END
80 IF CC$="ENTRY" OR LEFT$(CC$,2)="EN" THEN 1000
85 IF CC$="HELP" OR LEFT$(CC$,1)="H" THEN 6000
90 IF CC$="EDIT" OR LEFT$(CC$,2)="ED" THEN 2000
100 IF CC$="INDEX" OR LEFT$(CC$,2)="IN" THEN 300
0
110 IF CC$="GET" OR LEFT$(CC$,1)="G" THEN 4000
120 IF CC$="KILL" THEN 5000
130 IF CC$="END" THEN CLOSE:END
140 REM- AN INVALID COMMAND WAS ENTERED
150 REM- CLEAR SCREEN AND SHOW OPTIONS
160 PRINTCHR$(16);CHR$(22);
170 PRINT"OPTIONS ARE:":PRINT"COMMAND";TAB(15);"
ABREVIATION":PRINT"--------------------------":P
RINT"ENTRY";TAB(15);"EN":PRINT"INDEX";TAB(15);"I
N":PRINT"EDIT";TAB(15);"ED":PRINT"GET";TAB(15);"
G":PRINT"KILL":PRINT"END":PRINT"HELP";TAB(15);"H
"
171 PRINT"--------------------------"
180 REM- INPUT ANYTHING
190 PRINT:PRINT:INPUT A$
200 GOTO 60
1000 REM- INPUT ENTRY AND STORE AFTER VERIFICATI
ON
1010 REM- INPUT KEYWORD FIRST
1020 KK$="*":INPUT"KEYWORD";KK$
1030 REM- IS IT TOO LONG?
1040 IF LEN(KK$)>15 OR KK$="*" THEN 1020
1045 REM- CHECK FIRST CHARACTER OF KEYWORD FOR V
ALIDITY AGAINST STORED LIST
1050 RESTORE
1060 A$=LEFT$(KK$,1)
1070 DATA "A","B","C","D","E","F","G","H","I","J
","K","L","M","N","O","P","Q","R","S","T","U","V
","W","X","Y","Z","1","2","3","4","5","6","7","8
","9","0","END"
1080 READ A1$:IF A1$="END" THEN PRINT"INVALID KE
YWORD":GOTO1020
1085 IF A1$=A$ THEN 1090 ELSE 1080
1090 REM- NOW INPUT ASSOCIATED DATA
1100 EE$="*":PRINT"ENTRY FOR KEYWORD ";KK$
1110 LINE INPUT EE$
1120 REM- IS DATA TOO LONG?
1130 IF LEN(EE$)>110 OR EE$="*" THEN 1100
1135 F$=A$+"*"
1140 CLOSE:OPEN "R",#1,F$,0
1145 S=1:REM SET SECTOR COUNTER
1150 REM- USE MASTER FIELD STATEMENT
1160 GOSUB 9000
1165 REM- GET RECORD AT SECTOR S
1170 GET #1,S
1175 REM- CHECK IF DATA ALREADY IN SECTOR S
1180 IF RIGHT$(T$,1)="0" THEN S=S+1:GOTO1160
1185 REM- PUT DATA INTO SECTOR S, RETURN TO OPTI
ONS MENU
1190 LSET K$=KK$:LSET E$=EE$:LSET T$="100"
1200 PUT #1,S:CLOSE:GOTO 60
2000 REM- EDIT INFORMATION OF A PARTICULAR KEYWO
RD
2010 REM- NOTE-EDITTING OF KEYWORD NOT ALLOWED
2020 REM- INPUT KEYWORD
2030 KK$="*"
2035 INPUT"KEYWORD";KK$
2037 IF LEN(KK$)>15 THEN PRINT"TOO LONG!":GOTO20
30
2040 A$=LEFT$(KK$,1)
2050 REM- CHECK FOR VALID FIRST CHARACTER OF KEY
WORD
2060 RESTORE
2070 READ A1$:IF A1$="END" THEN PRINT"INVALID KE
YWORD":GOTO 2030
2080 IF A1$=A$ THEN 2090 ELSE 2070
2090 F$=A$+"*"
2100 CLOSE:OPEN "R",#1,F$,0
2105 REM- INITIALIZE SECTOR COUNTER
2110 S=1
2120 REM- GOSUB FIELD STATEMENT
2130 GOSUB 9000
2140 IF S=LOF(1) THEN PRINT"END OF FILE":CLOSE:G
OTO 60
2150 GET #1,S
2160 IF RIGHT$(T$,1)<>"0" THEN S=S+1:GOTO2130
2170 IF LEFT$(K$,LEN(KK$))<>KK$ THEN S=S+1:GOTO
2130
2180 REM- FOUND PROPER KEYWORD
2182 REM- CLEAR SCREEN
2185 PRINTCHR$(16);CHR$(22);
2190 PRINT:PRINTE$:PRINT:PRINT
2200 YN$="*":INPUT"WANT TO CHANGE THIS INFORMATI
ON";YN$
2210 IF YN$="YES" THEN 2225
2220 IF YN$="NO" THEN S=S+1:GOTO2130
2222 PRINT"USE YES OR NO":GOTO 2200
2225 PRINTCHR$(16);CHR$(22);
2230 EE$="*":PRINT"UPDATE INFORMATION":LINE INPU
T EE$
2235 IF LEN(EE$)>110 OR EE$="*" THEN PRINT"RE-EN
TER-TOO LONG":GOTO 2225
2240 LSET E$=EE$:PUT #1,S:S=S+1:GOTO 2130
3000 REM- PRODUCE AN INDEX
3010 REM- THIS ROUTINE GIVES YOU ALL
3020 REM- THE KEYWORDS WHICH START WITH
3030 REM- A SPECIFIED LETTER OR NUMBER
3040 PRINTCHR$(16);CHR$(22);
3045 PRINT"TYPE 'ALL' FOR TOTAL INDEX"
3050 A$="*":INPUT"INDEX FOR WHAT LETTER";A$
3055 IF A$="*" THEN 3040
3057 IF A$="ALL"  THEN 3300
3060 A$=LEFT$(A$,1)
3070 REM- CHECK FOR VALID INDEX
3080 RESTORE
3090 READ A1$:IF A1$="END" THEN PRINT"INVALID IN
DEX":GOTO3040
3100 IF A1$=A$ THEN 3110 ELSE 3090
3110 F$=A$+"*"
3120 CLOSE:OPEN "R",#1,F$,0
3125 REM- INITIALIZE SECTOR COUNTER
3130 S=1
3135 REM- RESET LINE COUNTER
3140 C=0:PRINTCHR$(16);CHR$(22);
3150 REM- GOSUB FIELD STATEMENT
3160 GOSUB 9000
3165 GET #1,S
3170 IF S=LOF(1) THEN PRINT"END OF INDEX":CLOSE:
INPUT A$:GOTO60
3175 IF RIGHT$(T$,1)<>"0" THEN S=S+1:GOTO 3160
3180 GET #1,S:PRINTS;TAB(5);K$:C=C+1:S=S+1
3190 IF C=14 THEN 3200 ELSE 3160
3200 INPUT N$;:REM- STOPS OUTPUT UNTIL YOU INPUT
 ANYTHING
3210 GOTO 3140
3300 REM- PRINT COMPLETE INDEX
3305 REM- INITIALIZE
3310 RESTORE:C=0:REM- C IS LINE COUNTER
3315 PRINTCHR$(16);CHR$(22);
3320 READ A1$:IF A1$="END" THEN PRINT"END OF IND
EX":INPUT A$:CLOSE:GOTO60
3330 REM- FORM FILE NAME
3340 F$=A1$+"*"
3350 REM- INITIALIZE SECTOR COUNTER
3360 S=1
3370 REM- OPEN FILE
3380 CLOSE:OPEN "R",#1,F$,0
3390 REM- GOSUB FIELD STATEMENT
3400 GOSUB 9000
3410 REM- END OF FILE?
3415 IF S=LOF(1) THEN 3320
3420 GET #1,S
3430 REM- DATA VALID?
3440 IF RIGHT$(T$,1)<>"0" THEN S=S+1:GOTO 3400
3450 REM- VALID DATA HAS BEEN FOUND
3460 REM- INCREMENT LINE COUNTER,DISPLAY LINE,IN
CREMENT SECTOR COUNTER
3470 C=C+1:PRINTS;TAB(5);K$:S=S+1
3475 REM- IS SCREEN FULL?
3480 IF C=14 THEN C=0:INPUT A$:PRINTCHR$(16);CHR
$(22);:GOTO3400
3490 GOTO 3400
4000 REM- GET ROUTINE
4005 REM- CLEAR SCREEN
4010 PRINTCHR$(16);CHR$(22);
4015 REM- INPUT KEYWORD
4020 KK$="*":INPUT"KEYWORD";KK$
4023 IF LEN(KK$)>15 THENPRINT"TOO LONG":GOTO 402
0
4025 A$=LEFT$(KK$,1)
4030 REM- CHECK FOR VALID KEYWORD
4040 RESTORE
4050 READ A1$:IF A1$="END" THEN PRINT"INVALID KE
YWORD":GOTO4020
4060 IF A1$=A$ THEN 4070 ELSE 4050
4070 F$=A$+"*"
```

permanently stored using the date with no year.

The type of information you choose to store and the keyword you use to summon information remains entirely up to you, with the following restrictions: 1) Keywords must begin with capital letters A through Z or digits 1 through 9. 2) Keywords cannot exceed 15 characters including spaces. 3) Entries must not exceed 110 characters.

Your bob-up program will support the following tasks:

1) Entry of keywords and associated data. The program checks keywords and data for conformance with the above restrictions before storing records on disk.

2) Editing of data associated with particular keywords. If you use a particular keyword more than once, the program lets you edit each entry. Before changing information stored on the disk, the program checks the new information for excessive length. NOTE – You cannot edit keywords.

3) Retrieval of an index of all keywords starting with a specificed letter. The program returns a record number and keyword for each entry. You must use this record number when killing any information. (See below.)

4) Getting the information associated with a particular keyword. The program checks each keyword for conformance with the above restrictions, then displays each entry with that keyword. After each entry is displayed, the computer waits for you to type "return" before searching for any other data stored under this keyword. After checking the file, the program informs you when there are no more entries.

5) Killing an entry and keyword. To do this, first obtain an index for the appropriate first letter or number of the keyword. The program then asks you for the keyword and record number of information you want to delete. This procedure avoids inadvertent erasure of data. The program displays the information in that record number and then asks if you want to delete the information. When you delete it, the storage space is freed on the disk, allowing another entry to be stored in this space.

6) End. This command closes all files and returns you to BASIC's command level.

The program includes appropriate error messages and prompts.

Output is formatted for a Southwest Technical Products CT-1024 Video Terminal. The line "Print CHR$(16); CHR$(22)" causes the cursor to home and screen to erase.

```
4080 CLOSE:OPEN "R",#1,F$,0
4090 S=1:REM- INITIALIZE SECTOR COUNTER
4100 IF S=LOF(1) THEN PRINT"END OF FILE":CLOSE:G
OTO60
4110 REM- USE MASTER FIELD STATEMENT
4115 GOSUB 9000
4120 GET #1,S
4125 REM- CHECK IF DATA IN SECTOR
4130 IF RIGHT$(T$,1)<>"0" THEN S=S+1:GOTO4100
4140 IF LEFT$(K$,LEN(KK$))=KK$ THEN 4160
4150 REM- THE KEYWORD DID NOT MATCH
4155 S=S+1:GOTO4100
4160 REM- CLEAR SCREEN,SHOW DATA
4165 PRINTCHR$(16);CHR$(22);
4167 PRINT"KEYWORD-";K$
4170 PRINT:PRINTE$:PRINT:PRINT:INPUT A$
4180 REM- CONTINUE SEARCHING FOR MORE INSTANCES
OF THE SAME KEYWORD
4190 S=S+1:GOTO 4100
5000 REM- KILL AN ENTRY ROUTINE
5005 REM- CLEAR SCREEN
5010 PRINTCHR$(16);CHR$(22);
5015 REM- INPUT KEYWORD
5020 KK$="*":INPUT"KEYWORD";KK$
5023 IF LEN(KK$)>15 THENPRINT"TOO LONG!":GOTO 50
20
5025 REM- CHECK FOR VALID KEYWORD
5030 A$=LEFT$(KK$,1)
5040 RESTORE
5050 READ A1$:IF A1$="END" THEN PRINT"INVALID KE
YWORD":GOTO 5020
5060 IF A1$=A$ THEN 5070 ELSE 5050
5070 S=0:INPUT"RECORD";S
5080 IF S=0 OR S>2000 THEN PRINT"INVALID RECORD"
:GOTO 5070
5085 F$=A$+"*":CLOSE:OPEN "R",#1,F$,0
5090 GOSUB 9000:GET #1,S
5095 IF LEFT$(K$,LEN(KK$))<>KK$ THEN PRINT"KEYWO
RD DOES NOT MATCH":CLOSE:INPUT A$:GOTO 60
5100 PRINTCHR$(16);CHR$(22);
5102 PRINT"KEYWORD-";K$
5105 PRINTE$:PRINT:PRINT
5110 YN$="*":INPUT"WANT TO KILL THIS RECORD";YN$
5120 IF YN$="YES" THEN 5150
5130 IF YN$="NO" THEN CLOSE:GOTO 60
5140 PRINT"USE YES OR NO":GOTO 5110
5150 GOSUB 9000:LSET T$="":LSET E$="":LSET K$=""
:PUT #1,S:CLOSE:GOTO 60
6000 REM- HELP SECTION
6010 PRINTCHR$(16);CHR$(22);
6020 PRINT"HELPFUL INFORMATION ON:"
6030 PRINT:PRINT"1.ENTRY OF DATA AND KEYWORDS"
6040 PRINT"2.EDITING OF DATA"
6050 PRINT"3.INDEX OF KEYWORDS"
6060 PRINT"4.RETRIEVING DATA"
6070 PRINT"5.KILLING OF DATA AND KEYWORDS"
6080 PRINT"6.ENDING THE PROGRAM"
6090 PRINT:PRINT"TO GET HELP ON ONE OF THE":PRIN
T"ABOVE SUBJECTS,TYPE ITS NUMBER ":INPUT"FOLLOWE
D BY 'RETURN'";A
6100 IF A<1 OR A>6 THEN 6010
6110 ON A GOTO 6200,6300,6400,6500,6600,6700
6200 PRINTCHR$(16);CHR$(22);
6210 PRINT"1.ENTRY OF DATA WITH KEYWORD"
6220 PRINT"COMMAND-ENTRY":PRINT"ABREVIATION-EN"
6230 GOSUB 6800:GOSUB 6850
6240 GOTO 6880
6300 PRINTCHR$(16);CHR$(22);
6310 PRINT"2.EDITING OF DATA"
6320 PRINT"COMMAND-EDIT":PRINT"ABREVIATION-ED"
6330 PRINT"TO EDIT DATA,INPUT ITS KEYWORD."
6340 GOSUB 6800
6350 PRINT"YOU WILL BE ABLE TO REVIEW THE":PRINT
"PRESENT DATA, AND IF DESIRED,":PRINT"REPLACE IT
 WITH NEW DATA."
6360 GOSUB 6850
6370 GOTO 6880
6400 PRINTCHR$(16);CHR$(22);
6410 PRINT"3. INDEX OF KEYWORDS"
6420 PRINT"COMMAND-INDEX":PRINT"ABREVIATION-IN"
6430 PRINT"TWO TYPES OF INDEXES ARE":PRINT"AVAIL
ABLE. ONE IS AN INDEX OF":PRINT"ALL KEYWORDS BEG
INNING WITH":PRINT"ANY LEGAL CHARACTER."
6450 PRINT"ALSO, AN INDEX OF ALL KEYWORDS":PRINT
"IS AVAILABLE BY TYPING 'ALL'":PRINT"WHEN YOU AR
E ASKED WHICH INDEX":PRINT"YOU WANT."
6460 GOSUB 6870:GOTO 6880
6500 PRINTCHR$(16);CHR$(22);
6510 PRINT"4.RETRIEVING DATA"
6520 PRINT"COMMAND-GET":PRINT"ABREVIATION-G"
6540 PRINT"TO RETRIEVE SOME DATA,":PRINT"YOU INP
UT ITS KEYWORD."
6560 PRINT"ALL STORED ENTRIES WITH THIS":PRINT"K
EYWORD WILL BE DISPLAYED."
6570 GOSUB 6870
6575 PRINT"WHEN ALL STORED ENTRIES ARE":PRINT"CH
ECKED, THE MESSAGE 'END OF":PRINT"FILE' IS DISPL
AYED."
6580 GOTO 6880
6600 PRINTCHR$(16);CHR$(22);
6610 PRINT"5.KILLING OF DATA AND KEYWORD":PRINT"
COMMAND-KILL":PRINT"NO ABREVIATION"
6620 PRINT"TO KILL AN ENTRY WITH ITS ":PRINT"KEY
WORD, FIRST GET ITS RECORD":PRINT"NUMBER FROM AN
 INDEX."
6630 PRINT"THEN YOU INPUT ITS KEYWORD AND":PRINT
"RECORD NUMBER. A RECORD NUMBER":PRINT"MUST BE B
ETWEEN 1 AND 2000."
6640 PRINT"THE MESSAGE 'KEYWORD DOES NOT":PRINT"
MATCH' WILL BE DISPLAYED IF":PRINT"THE STORED KE
YWORD OF THE REC-":PRINT"ORD-YOU SPECIFY DOES NO
T MATCH"
6650 PRINT"THE KEYWORD YOU INPUTTED."
6660 GOTO 6880
6700 PRINTCHR$(16);CHR$(22);
6710 PRINT"6. ENDING THE PROGRAM":PRINT"COMMAND-
END":PRINT"NO ABREVIATION"
6720 PRINT"TO CLOSE ALL FILES AND EXIT ":PRINT"T
HE 'BOB-UP' PROGRAM, USE THE":PRINT"'END' COMMAN
D."
6730 GOTO 6880
6800 PRINT"A KEYWORD IS ANY WORD UP TO 15":PRINT
"CHARACTERS LONG, INCLUDING  ":PRINT"SPACES.IT M
UST BEGIN WITH ":PRINT"AN UPPER-CASE LETTER FROM
 A-Z":PRINT"OR A DIGIT FROM 0-9."
6805 RETURN
6850 PRINT"DATA ASSOCIATED WITH A KEYWORD":PRINT
"MUST BE NO LONGER THAN 110":PRINT"CHARACTERS, I
NCLUDING SPACES.":RETURN
6870 PRINT"WHENEVER A '?' IS DISPLAYED,":PRINT"T
HE COMPUTER IS WAITING FOR ":PRINT"YOU TO TYPE '
RETURN' BEFORE ":PRINT"PROCEEDING.":RETURN
6880 YN$="*":INPUT"NEED MORE HELP";YN$
6890 IF YN$="YES" THEN 6000
6895 IF YN$="NO" THEN 60
6898 PRINT"USE 'YES' OR 'NO'.":GOTO 6880
9000 FIELD #1,3 AS T$,15 AS K$,110 AS E$:RETURN

OK
```

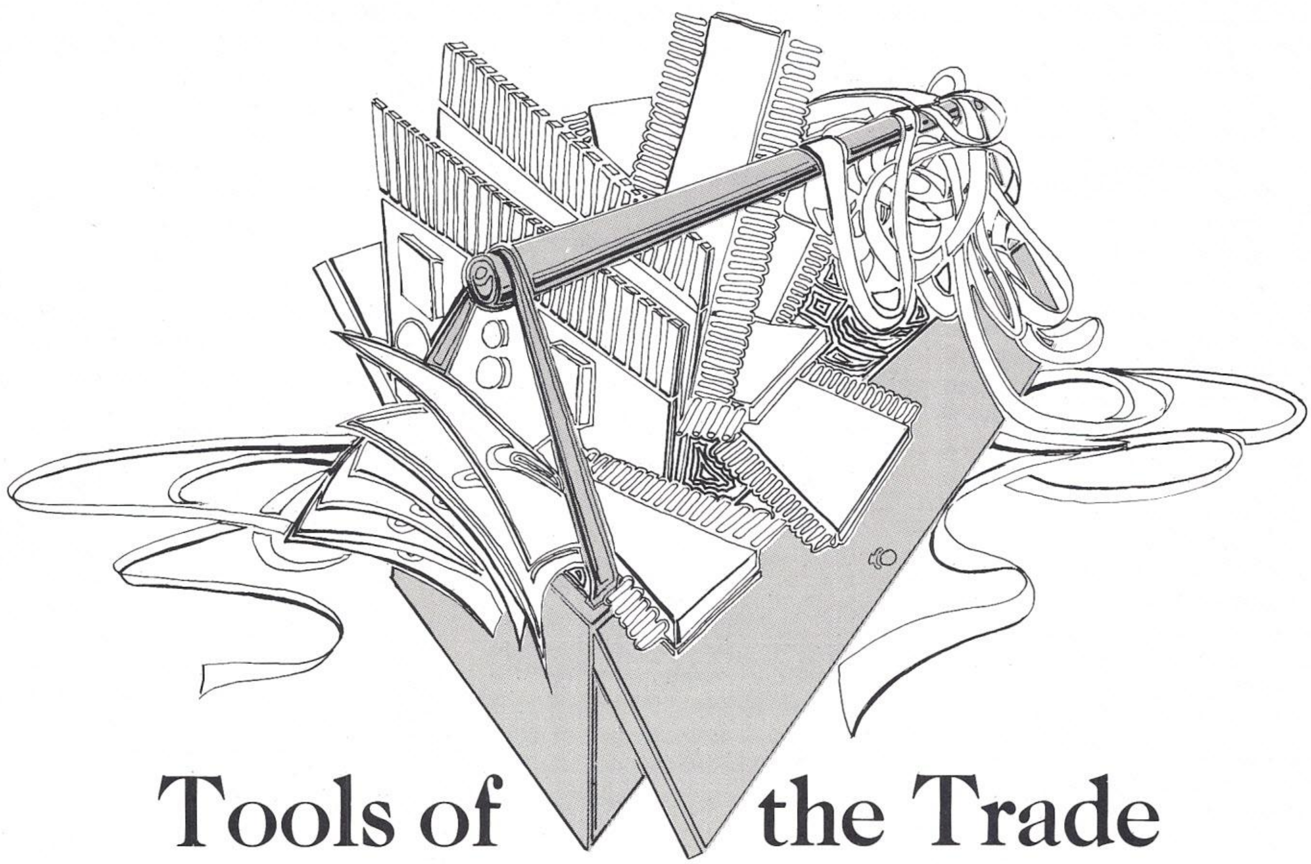
Program Notes

1. Including instructions and remarks, the program takes 8200 bytes of storage.

2. When using the "get" command to retrieve information, the program only checks the stored keywords up to the length of the keyword you input. That is, if you input a keyword "D" to search for, all entries beginning with "D" will be displayed. If you input "CA" as your keyword, all entries with keywords starting with "CA" will be displayed. To see all entries for April, for example, input keyword "4".

3. The "HELP" command will give you some information on the use of any of the six program commands. It is formatted for a CT-1024 screen.

COMPUTER LANGUAGES



Tools of the Trade

BY JEF RASKIN

Choosing a "universal" or "best" computer language can be a trap. While many people accept BASIC as the "be all and end all" of computer languages, others see different languages as tools fitting different needs.

There is really only one higher level language that you can be serious about if you have a personal computer: BASIC. All microcomputer makers have chosen to offer BASIC and nobody has offered anything else yet. Assemblers are not higher level languages. FORTRAN, APL, and FORTH are becoming available, but *not* from the hardware manufacturers. Let's talk about BASIC first.

The best thing about BASIC is that it's here. It is widely used, easier to learn than some other languages and is usually *interpreted.* Interpretation is one of the two fundamental ways to make a computer understand a higher level language. The other method is *compilation.* And there are lots of in-betweens.

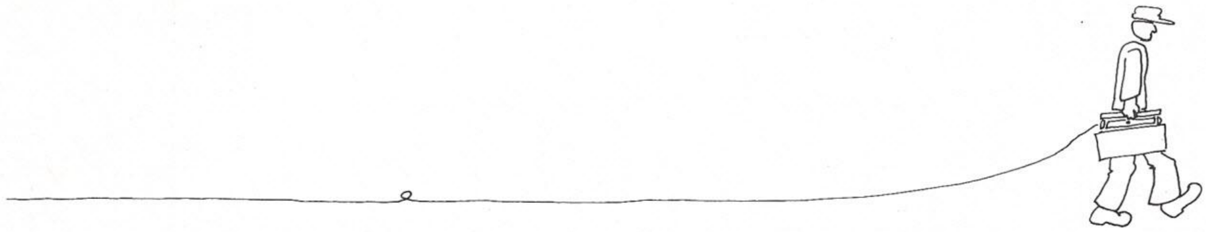
Compilation is the more obvious of the two methods, though it's usually harder to implement. The computer works in some *machine language,* which is language that it was built to understand. Machine language is tedious to program in. A *compiler* is a program that takes a program written in some higher level language such as FORTRAN or ALGOL and *translates* it into machine language. In a separate step the computer *executes* language.

An *interpreter* is a program that operates on a program written in some higher level language, such as BASIC or APL, and performs the operations. The BASIC program is executed directly.

With a compiler the original program written in, say, FORTRAN is called a *source* program. The machine language translation is the *object* program (often called *object code,* "code" being a synonym for "program"). With an interpreter there is no object program – just source code.

Interpreters and compilers can be written in any language. It is possible

Illustrations by David Gardner



to write a compiler for ALGOL in BASIC or an interpreter for APL in FORTRAN. In practice most interpreters and compilers are written in assembler code.

An *assembler* is a kind of compiler, in that it translates from some source language to machine language. The source language in this case is designed for a *specific* computer and has a flavor close to machine language. Higher level languages, by contrast, are usually intended to work on almost *any* machine.

Application programs include games, accounting, scientific calculation, graphics – anything that the computer does at the user's behest – and are far easier to write in higher level languages than in assembler. The programs that help the user run the computer – the assemblers, compilers, interpreters, monitors and the like – are called *system* programs.

So assemblers are designed for system programming on a particular computer. There is at least one assembler for each type of computer. Higher level languages are designed to make particular kinds of *tasks* easier without regard for the kind of computer that will carry out the tasks. The differences between the tasks that higher level languages were designed to do accounts for much of the variety found among those languages.

So we can separate languages into two piles: Those that are interpreted and those that are compiled. While *any* language can be treated either way, we can safely put BASIC, APL and FORTH into the interpreted camp.

BASIC is not a very powerful language: Although simple programs may be easily and quickly written in BASIC, more complex and involved programs are relatively difficult. There are many ways in which BASIC is inherently a weak language for complex problems. One problem is the limited names for variables, which in most BASICs can only be a character or two long. It's hard to remember, in a program that might require dozens or hundreds of variables, which variable does what function if only unmemorable names are available. A few newer BASICs allow longer names, a definite improvement.

Trying to write a complicated program is like trying to build a large house by yourself. You can do the task, but more easily if you can break it down into smaller tasks independent of each other. While installing the plumbing, it is nice not to have to keep in mind the catalog number of the nails you used in the room a month earlier. Large scale programming is like that: It is a bother to remember if you used T5 (or was it T6 or M5?) to represent a particular quantity, such as the monthly totals in an accounting program. A name such as MONTHLY-TOTALS would be a lot easier to remember. If BASIC had subroutines where the variables were *local*, the programmer wouldn't even have to know what names were used in any other part of the program. Defining variables for only particular areas of a program allows different programmers to cooperate on a large project, without having to know the details of each other's portions.

APL is a different matter. Whereas BASIC was designed by a couple of mathematicians for students, APL was designed by a mathematician for programmers. BASIC stands for "Beginners All Purpose Symbolic Instruction Code". APL stands for "A Programming Language". That spells out the difference.

APL is a very powerful language in which programming almost any application is easier than in BASIC (or almost any other language). When talking to an experienced user of APL one often feels in the presence of a fanatic. It is well liked by programmers who have learned it.

Several implementations of APL are reportedly in the works for personal computers. Unfortunately, APL requires a special character set.

FORTH is another very unusual language. Available from FORTH, Inc., for microprocessors, it is very unlike most higher level languages. Whereas most higher level languages run significantly slower and/or require more memory space than does a well-coded assembly program, FORTH programs tend to be quite fast and take less space than an equivalent program written conventionally. FORTH was originally intended for process control, but has proven to be widely applicable.

Both APL and FORTH are very unlike the "mainstream" style languages that tend, at least superficially, to look somewhat alike. These include ALGOL, FORTRAN and PL/1, and their manifold variations or *dialects.*

ALGOL, used more in Europe than in the United States, is (like FORTRAN) a very old and seminal language. Intended mainly for scientific applications, it is available on very few minicomputers and is not yet available on any microcomputers. Programs in ALGOL are broken up into *blocks,* and subprograms are called *procedures.* ALGOL has permeated the newer languages, but little of it has percolated through to BASIC. "ALGOL" stands for ALGOrithmic Language.

FORTRAN, designed for scientific applications (the name stands for FORmula TRANslation), is the most widely used scientific computer language in the world. As FORTRAN becomes available on personal computers, a wealth of programs written over the last couple of decades will become useable by microcomputer lovers.

So far I've treated BASIC as though there were such a thing. Sorry. There is no such language as BASIC.

That may seem a bit strong, but there *is,* for example, a standard COBOL and FORTRAN, defined by ANSI (American National Standards Institute) but ANSI's BASIC is newer and hasn't found wide acceptance yet. Thus a BASIC program written for a Polymorphic 88 won't necessarily run on an APPLE II or a SOL or on GE Timesharing or on a PDP-11. No two BASICs are quite alike. But they are usually similar enough so that having learned one you can glom onto their differences in a matter of minutes. The differences are just great enough to be

The carpenter does not insist that one tool do all jobs, but chooses the appropriate instrument from among those on hand. So should you.

annoying when moving from one machine to another.

There is an old saying that a camel is a horse designed by a committee. The very large and cumbersome language PL/1 seems to fit that description of a camel. PL/1 is a melting pot of the features of ALGOL, FORTRAN, COBOL and bits and pieces of everything thrown in without regard to elegance. It requires rather large computers, so it won't be seen in home systems for a while – if ever.

Nobody ever uses all of PL/1. It has so many features that programmers tend to use just a convenient subset of it. Of course each programmer chooses a different subset, but that's just a corollary of Murphy's Law. The name means "Programming Language One", which is a bit pretentious, since some two hundred programming languages preceded it.

COBOL, the COmmon Business Oriented Language, was designed (if you haven't guessed) for business applications. It tends to make report generation easier, and is not too difficult to read. On the other hand, it is a verbose language and a pain to enter into a computer. COBOL was also designed for larger machines and while it is widely used, many installations find APL or BASIC handier. It has narrow application areas and is not expected to make a debut on microcomputers in the near future.

The standard language for much Artificial Intelligence work is LISP. This language is eccentric – heavily bound with parentheses and a terminology no where near any other. In its appropriate place it has many staunch admirers. In fact, the weirder a language is, the more strongly its users cling to it.

SNOBOL is a string-processing language made for analyzing texts, processing words, and generating concordances. In use mainly at universities, its popularity is waning. But study its features if you want to process strings, since it was well designed for that.

PL/M is like PL/1, but trimmed of most of the fat and designed for microcomputer use. It is expensive and most versions run only on very large microprocessor systems. But PL/M is powerful and suitable for almost any application you might wish to do with microcomputers. Many professional programmers use PL/M, but few hobbyists have access to it yet.

Another language coming into its own is PASCAL, named for the mathematician Blaise Pascal. It is a structured language designed for teaching structured programming. It is elegant but tends to take up more than 20 kilobytes, so it won't be a big seller in a world where the average machine size is 16 kilobytes. As memory prices continue to drop, though, languages like PASCAL will become more prominent.

Tiny BASIC is important on the home computer scene. Typically, Tiny BASIC only allows integer arithmetic, includes only the most necessary statement types and has few bells and whistles. Tiny BASIC is interesting not because of its power, which is miniscule, but because it is compact. Since versions of Tiny BASIC require as little as 2½ kilobytes, they are well suited to even the smallest hobbyist systems.

Besides compilers, assemblers and interpreters (and the many systems that fall somewhere between these narrow categories) there are also *cross-assemblers* and *cross-compilers.* These terms have nothing to do with Christian theology, but simply refer to an assembler or compiler that runs on a computer of type A and produces object code for a computer of type B. A is usually a large computer and B a small one. Since most FORTRAN compilers are too big to fit on most microcomputers, it is common to design a FORTRAN compiler that runs on a big computer and produces code for a small one. But since cross-compilers are handy only if you have access to *both* computers, they don't interest the personal computerist.

That's just a sampling of the many computer languages. In the thirty years that computers have been around, over three hundred different languages have been invented for them. This count includes languages like WATFOR and WATFIVE, which are dialects of FORTRAN. Expose yourself to as many languages as possible, since each language gives you a different view of how to organize tasks.

Don't get caught in the trap of asking which language is best, or whether some language should be made the universal standard. Languages are our tools. A hammer won't do where needlenose pliers are called for. Similarly, different computer languages fit different needs.

The carpenter does not insist that one tool do all jobs, but chooses the appropriate instrument from among those on hand. So should you.

It is sad that most personal computer buffs have only the crude tool of BASIC and awkward and hard-to-handle assemblers to work with. I've used all but two of the languages described in this article, and taught most of them as well. Given an unseen task to program as easily as possible, I'd usually choose to do it in APL. The text-editing program that writes these articles on my home computer is in BASIC, because it's all that was supplied and I haven't had time to write an interpreter for some other language. Most of my programs were in FORTRAN because it was available. Programs suitable for assembler-level work I'd prefer to do in FORTH; but since a copy of FORTH costs $1000, it will be a while before I choose to use it.

Having taught different languages to thousands of students, I can affirm that BASIC is not the only language easy to teach. Any language that can be executed and modified immediately from a terminal is easy to teach, and any language whose implementation requires considerable delay and frustration is hard to teach. A logically straightforward language like APL, if presented well, is not at all difficult for a beginner to grasp. Too bad that APL requires a special character set to be effective. I suspect that the future will show many languages designed especially for microcomputers. Keep an open mind.

Diving into Computer Advertising

BY GARY GREENBERG

Most computer novices require about six months to a year of background reading before they're competent enough to analyze computer ads in magazines and spend their money wisely. If you fall into this category, then you've probably fought the temptation to rush right to the nearest post office and order a system you just saw advertised. But beware; hasty decisions can cost you thousands of dollars in unsatisfactory equipment.

Ads can be vague. They do not always clearly describe the capabilities of the advertised product. Here are a few guidelines to help you analyze ads so you'll end up with a system that fits your needs.

Rule number one: never buy a system or product merely on the basis of an ad. Most worthwhile systems and products are subject to evaluative articles in the various hobby magazines. Always read these articles first. Furthermore, most companies provide literature that goes into greater detail about the product than the ads. Always ask a company for this extra bit of literature; it can't hurt. Some companies will request nominal fees for the more detailed information, but pay it – invariably the information you buy will provide you with an education on hobby computers.

If you can't obtain any additional research material, talk to friends, dealers or computer club members about the item. Without such backup research, I would be careful about spending my money.

Do not assume that an ad for "a complete computer system" will provide a computer to meet all your needs. Many advertised systems require substantial expenditures necessary to upgrade the system before it can operate.

To determine the type of computer system you need, you must consider a number of things. For one, do your interests lie in hardware or software? Hardware buyers will probably be satisfied with the cheaper kits rather than the assembled systems. A lot of memory will not be a high priority, nor will a high level language

capacity. However, the variety of peripherals available and how to build them will be important. The initial interest here is in getting a system up and running rather than writing a new version of Star Trek.

Low cost evaluators are a good buy for the hardware person; but before selecting such evaluators be aware of expansion possibilities and peripheral availability. If you're a software fan, you'll want to know how much memory you'll need for running either purchased programs or for personal creations. Other considerations not to lose sight of include: ease of input, program and data storage, and speed of execution. You can't run BASIC with only 1 or 2K of memory. (Did you know that?) 4K will permit use of Tiny BASIC and allow some room to write your programs. However, Tiny BASIC is a greatly limited form of BASIC and many people will want a more powerful BASIC.

The next level up is a 4K BASIC. This version of BASIC is fairly powerful, but usually lacks string and array manipulation capabilities. To use 4K BASIC you will need at least 6 to 8K to write the program and store the BASIC language in the computer. As a general rule, 4K of unoccupied memory will permit insertion of 100 lines of program.

The largest BASIC utilizing a cassette storage system is usually 8K BASIC. Designations such as 4K, 8K and 12K don't refer to the exact length of the BASIC language code. They're only an approximation of size, but tend to be a good guide to the capabilities of the language. Always assume you will need a minimum of 4K of memory above the length of the BASIC to write programs. Some 4K BASICS may occupy from 3.5 to 5K of memory. So, keep that in mind when designing your system. The 8K BASICS usually contain substantial string and array capabilities and frequently contain some file operation commands as well. If you plan to do file manipulations check out the commands available in your BASIC package.

Moving up to 12K BASIC usually means you'll use a disk-based storage system. If you plan to write any business ledger type of software and if you use disk BASICS (12K), then you'll probably need 32-48K of memory.

Your concern with program storage capability centers around two main areas. The first involves the ability to save a program for later use. Any program stored in the computer's random access memory (RAM) will disappear when the computer is turned off. If you don't want to re-enter your program every time you turn off the computer you'll have to preserve it in some manner.

The second reason for storage capability to space. Even if you leave your computer on full time, you'll soon run out of space to store your programs.

Some popular storage methods include ROM chips, audio or digital cassettes and floppy disks. Usually the monitor and housekeeping functions are stored in ROM (Read Only Memory). ROM can't be used for program writing. ROM does store higher level languages such as BASIC. When BASIC is stored in ROM you don't have to reload the program every time you turn on the computer. On most computers, at present, you have to feed in the BASIC language before you can write a BASIC program on the computer. With a floppy disk, this allows instantaneous execution. A cassette, however, could require several minutes of execution time for loading.

Audio cassettes are the most popular form of storage for hobby computers because of their simplicity and low cost. A C-30 cassette can hold about 200K bytes of information. Its major drawbacks include slow transfer rate and serial storage form. A cassette usually transfers data at the rate of 300 baud to 1200 baud, which translates to about 30 to 120 bytes per second. And with serial storage you must pass through the entire tape to reach the point you need.

Some systems permit the computer to start and stop the forward control while reading the data on the tape. Other systems require manual control of the tape. In the latter case, you can use a cassette with a tape counter to make the location of desired information, or you can use recording facility to verbally identify the information that follows the voice.

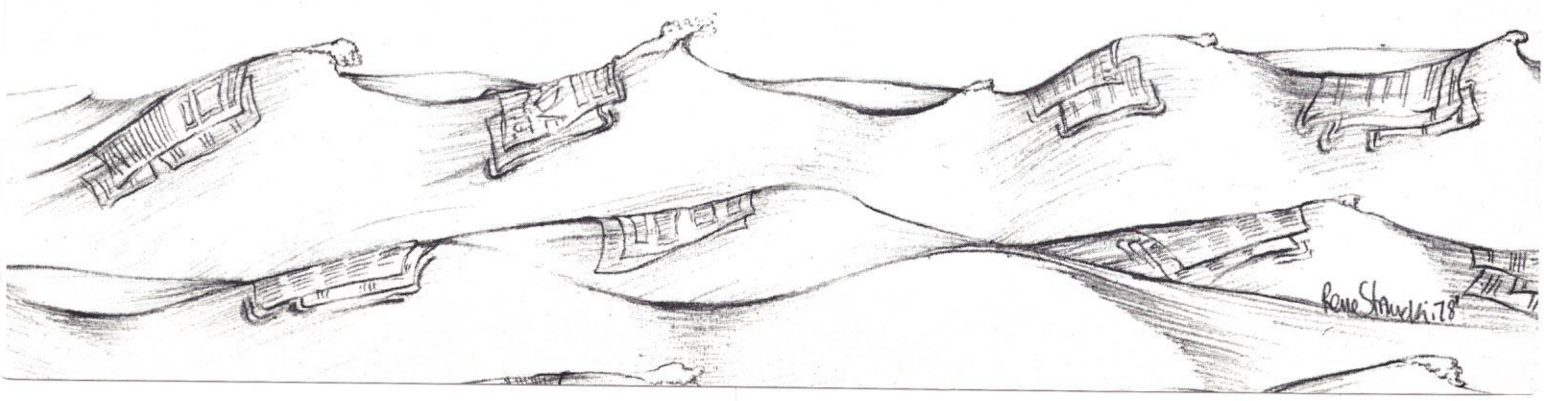
Although they cost more, digital cassettes are much faster and more reliable than audio cassettes. When you buy a digital system the manufacturer will frequently provide software to control the forward and backward controls so that you can search the tape for data – check to see if the software is included.

Floppy disks are another expensive device for storing data. However, they're even faster than digital cassettes and permit random access to the data. Think of the distinction between serial and random as you would the difference between locations of the fourth song on a cassette and fourth song on a record album. On the record you can place the needle directly on the fourth band. That is random access.

For a comparison of the three devices, consider average access times: floppy disk, 1/3 second; digital cassette, 1/2 minute; audio cassette, 2 minutes.

If you plan to use your computer for a filing system, then make sure your system can utilize two mass storage devices such as a dual floppy system or a dual cassette system. Many systems include a single cassette interface and you can not add a second cassette. Make sure your software contains the necessary commands to utilize filing controls. Also, be aware of the baud rates utilized by the system. A 300 baud controller can't run 1200 baud taped software. Some systems permit use of several different baud rates.

If an ad refers to a hex keyboard, remember you'll only be able to input those numbers expressed in hexadecimal notation – which is worthless if you plan to program in BASIC. To input BASIC you need an alphanumeric keyboard utilizing at least the upper case ASCII standard characters along with numbers and punctuation. When reading an ad, check for desired features. If not mentioned, as-



sume they're not included. Many applications, for example, require lower case alphabet. But not all keyboards permit lower case.

If you see an ad offering a "complete computer system" for only $250 with keyboard display and cassette interface, you can be sure (for now, at least) that they're talking about a hex keyboard and an LED display. The LED usually displays a single line of information consisting of a few numbers that can not be used for BASIC.

A popular display is a television-type screen or monitor, usually referred to as a CRT display. Frequently, the CRT is combined into a single unit with the keyboard and video interface. This is the TV terminal. Many ads for terminals include only the keyboard and video interface. The monitor portion is extra. Before you order make sure you know what is included. Frequently, an ad will quote some attractive price along with a picture including the CRT. Closer examination, however, usually reveals that the CRT is not included in the spiffy price.

When choosing your CRT there are a number of options to consider. Unless the ad says otherwise, assume your display area will be 16 lines by 32 columns. If that is not sufficient for your purposes, make sure you specify a bigger display before you order. (The next standard size is usually 16 x 64.) Other combinations often available on some of the higher priced terminals are 24 x 80 and 40 x 80. Eighty columns across corresponds to the line length on your 8½ inch wide paper. Other options to consider include lower case letters, scrolling and a cursor that responds to software control.

Naturally, every addition costs extra. The standard terminal, with upper case only, CRT, keyboard, video interface and 16 x 32 display will cost a minimum of $500.

Printers are a more expensive form of display than CRTs, but they provide permanent readable print-outs of unlimited length for your program. Before deciding which computer to buy, consider the column width you'll want. Different models offer different numbers of columns as well as different speeds the computer can print (lines per minute, characters per second).

You can choose a dot matrix printer or an impact printer (forming solid line characters). The dot matrix will be substantially cheaper but does not provide the type of printout needed for quality work.

Lower case is another consideration, as are the interface connections. Check whether the computer is configured to suit your needs. The printer may have a parallel interface while your computer utilizes only the serial interface.

Expansion should be an important consideration. Many systems are incapable of expanding to suit your purposes, so buy right in the beginning.

One immediate choice involves what bus to use. The S-100 bus is the major hobby standard; but even there, not all peripherals advertised as being S-100 compatible work with all S-100 systems. However, there are more peripherals and competing producers of similar peripherals for the S-100 than for any other bus standard. S-100 systems are usually organized around the 8080 and Z-80 chips.

If you reject the S-100 bus you're going to run into some problems. Most companies create bus dimensions that are incompatible with competitors' products. However, there does seem to be a movement developing for a SWTP bus designed around the 6800 chip. Also, some product interchange is going on now between companies.

If upward expansion is necessary when you leave the S-100 bus you'll have to check the catalogues of available peripheral support. Apparently, there are very few non-company options around. Check not only for variety and quality but also for existence. Many announced peripherals have not been produced yet.

Speculation exists concerning companies trying to support the IEEE-488 bus used by Commodore's PET. But at this writing no products are available, even from Commodore.

At a recent computer fair, the Commodore people said if you wanted a sound recorder you would have to use their cassette recorder – but even that isn't yet available. They also hinted that another company is already at work on a floppy disk for the PET.

Many ads offer software packages including BASIC and Operating Systems – but these packages are not always interchangable between computers and they often require some modification. And, in come cases, modifications aren't even possible. Always check to see how compatible your system is with someone else's.

Many ads flaunt the powerful instruction sets of its computer's chip. However, this is mainly of interest to persons who want to program in machine or assembly language.

If you plan to program in BASIC, which chip you use becomes irrelevant, although 6502-designed BASICs tend to be among the fastest. The main consideration, if you are planning to program in BASIC, is not which chip to use, but the capabilities of the BASIC instruction set.

When you see an ad mentioning software as a support to a particular chip (for example, 6800 Editor), the ad is referring to machine or assembly language, not BASIC.

On the average, assume that a system with 16K of memory, 8K BASIC, keyboard, CRT, a cassette and appropriate interfaces and connections will cost a minimum of $2200. Some systems can be put together for less, especially if you utilize kits. There are a couple of complete systems available for between $600 to $1500, assembled. However, they're not S-100 oriented and upward expansion has to be considered. Most of these less expensive systems utilize the 6502 chip and have demonstrated some of the fastest and most powerful BASICS available.

If an ad offers a peripheral at an unusually low price, it will usually be available only in kit form and frequently the kit doesn't include the cost of interfacing.

The "complete computer for a very low price" often also does not include a video monitor. If it is a bare-bones system, look for the cassette interface. Check to see if it is a kit or assembled model – very little memory is usually included in the quoted price.

In conclusion let me leave you with a final warning. The existence of an ad, with pretty pictures of the product, does not mean that the product exists or is available for delivery. Many companies, including some of the biggest hobby computer companies, frequently advertise a product that is still in the development stage.

Try to find out how many of the advertised products have been delivered.

Other companies, usually newer ones, advertise for months, accept orders and still fail to come up with a finished product. You might have to wait several months for your computer only to find that none are available and there aren't going to be any.

Although speed is one of the main qualities of a computer, slowness of delivery is one of the most common problems in the industry. So, when you're ready to jump in, hurry up and wait.

Speech Synthesis Makes Computers Talk

BY MICHAEL CHESTER

Circuitry consisting of a pulse source, a noise source and several resonators can generate all the sounds of human speech. This analog circuitry models the voice tract, with a small set of parameters specifying the states of the system. In a system developed by Computalker of Santa Monica, CA, the parameter set consists of four amplitudes and five frequencies.

The most fundamental kind of speech synthesis, referred to as "synthesis-by-rule", translates the sounds of speech (phonemes) into values for the required circuit parameters. These phonemes consist of the various consonant and vowel sounds that make up a spoken language. Therefore, the synthesis program must define the circuit control parameters corresponding to each phoneme.

Suppose you want the system to say a particular sentence. After you put the sentence into the synthesizer as a string of phonemes, the synthesizer converts each phoneme into the appropriate set, and the computer speaks the sentence – perhaps with a "Martian" inflection, but in words recognizable to the listener.

Methods of Synthesis

Let's look into what different developers have done with their speech-by-rule synthesizers. Model 1000, developed by Ai Cybernetic Systems, employs 64 phonemes, each represented as 16 control bits stored in a ROM location. These 16 bits determine the amplitudes, frequencies and durations of the analog parameters.

Model 1000's input coding, based on an ASCII subset, uses letters, numerals and some punctuation symbols in a six-bit-per-character code for accessing the 64 phonemes. In this one-character-per-phoneme arrangement, Ai Cybernetic tried to make the input character suggestive of the resulting sound. With most consonants this method is straightforward; for example, "K" stands for the usual k-sound that we're used to. Many English vowel sounds aren't that well-behaved; nonalphabetic symbols are matched to sounds in associative ways, such as the symbol "&" for representing the "a" sound in the word "and", or the symbol "#" for the "er" sound as in "number".

To synthesize, the user first programs a desired string of ASCII characters; the driver program then takes the characters one at a time, accesses each appropriate phoneme, checks a busy bit and transmits the required information to the sound-producing circuits. Busy bits are necessary, because the transmission rate per phoneme varies considerably. Model 1000 software uses a modest amount of programming – less than 50 bytes of 8080 machine code or about half a dozen lines of BASIC, at a transfer rate of less than 50 bytes per second.

For a typical example of the input to the synthesizer, suppose you want the system to say "I am talking to a robot." Then you would enter:
P$="&&IE AM AE T) . . KEN-RO.B) ..T

Votrax also developed more sophisticated and more costly systems employing these same general principles. The VS-6 uses an 8-bit command word, with 6 bits accessing a set of 61 phonemes and 2 bits determining inflection. A more advanced Votrax model, the ML-1 (with "ML" standing for "Multi-Lingual"), extends the system to 122 phonemes. It uses a 12-bit command word with 7 bits specifying the phoneme, 3 bits the inflection and 2 the duration. The "multilingual" capability means an ability to synthesize German, with expansion to other lan-

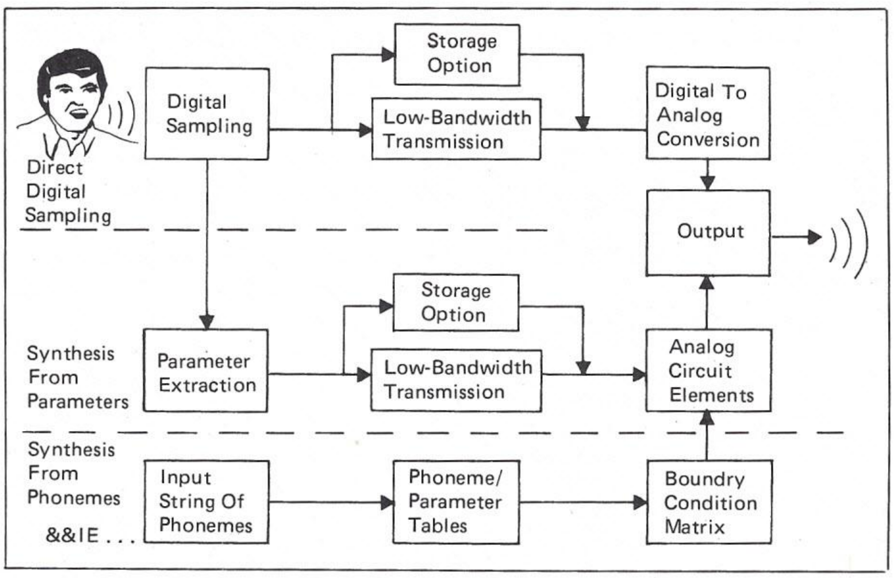


Fig 1 These three methods of producing speech share some common system elements but yield different results. Direct sampling speech sounds clear, but uses a limited vocabulary. Synthesis from parameters, another synthesis technique with limited vocabulary, produces speech with less clarity than direct sampling, but requires less memory. The speech generated by synthesizing from phonemes sounds "Martian," but the method allows use of an infinite vocabulary.

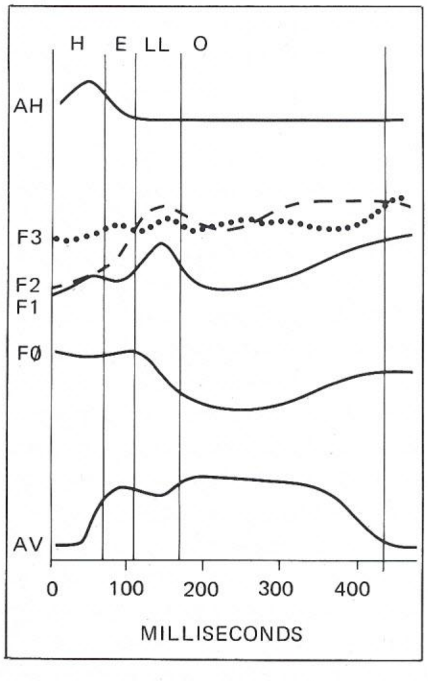


Fig 2 In parameters extracted from human speech, only six of the nine parameters vary in the word "hello." The full set of parameters includes voicing amplitude (AV), voicing frequency (FO), formant 1 frequency (F1), formant 2 frequency (F2), formant 3 frequency (F3), aspiration amplitude (AH), frication amplitude (AF), frication frequency (FF) and nasal amplitude (AN).

guages (Spanish, French, Japanese and Parsi) under consideration. The expanded phoneme set makes the sounds unique to specific languages. Both synthesizers are programmed to provide natural transitions between consecutive phonemes.

Another company producing sophisticated voice synthesis devices, Interface Systems has named its system DAVID (Digitally Actuated Voice Information Device). The user can program the synthesizer over a Bell System Touch-Tone TM telephone by dialing DAVID's telephone number. The synthesizer answers the call with "Hello, this is DAVID." It then asks for your password; after you have entered it, the synthesizer offers you a choice of demonstration programs. After choosing your program, you continue to interact with the computer in real time, entering numerals, * and # symbols and letters. You code a letter by pressing the letter button on your telephone, followed by 1, 2 or 3 to indicate one of the three letters on that button. What you get is spoken output. In other words, the Touch-Tone telephone becomes a simple computer terminal. Other speech output systems also offer Touch-Tone options – for example, Votrax's speech synthesis and digital sampling systems.

Computalker Consultants developed a phoneme-synthesis approach that stores the digital parameters in software, in contrast to the usual practice of setting the parameters permanently in solid-state memory. By keeping the definitions flexible, Computalker can easily reprogram phonemes to provide variations in emphasis, expression and pitch, and also to maintain a continuous updating in speech quality. This flexibility does make the software less compact. The synthesis-by-rule software package CSR1, which can be called from a keyboard or from a user program, requires nearly 6K bytes of memory in 8080 assembly code. Computalker's system includes a matrix for transition between any two phonemes for adjusting each phoneme to reflect boundary perturbations in which the phoneme is modified by its neighbors.

Whether synthesis from phonemes is set in ROM or left variable in software, it stands in contrast to the storage of actual human speech by digital sampling. Since the sampling technique is essentially a form of recording, it produces much clearer voice characteristics. However, the messages that the computer can produce are limited to those that were originally fed into memory. A phoneme-synthesizer can say anything, even though its voice tends to be a peculiar mixture of child-like and robotlike speech (perhaps because these systems are still learning to talk).

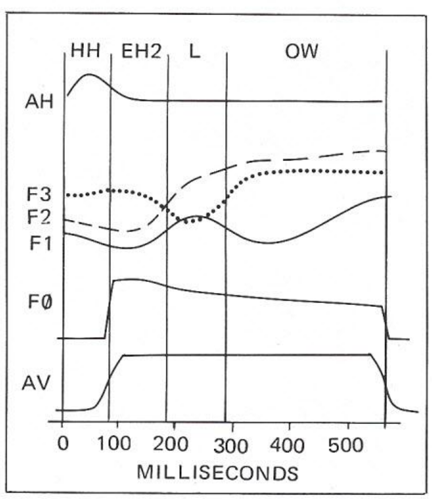


Fig 3 In parameters generated by rule, notice that one or two letters represent phonemes in this system. The "2" after phoneme EH indicates stress level for that phoneme.

On the other hand, digital sampling advocates see the trend toward increasingly compact, lower-cost memories as a way of providing such large vocabularies that a finite store won't seriously limit the number of messages possible.

A compromise approach extracts analog parameters from digitized speech; it still employs a finite-vocabulary, but it also loses some quality compared with direct sampling. However, it requires much less storage than direct sampling. Computalker uses parametric synthesis as an alternate mode, with the extracted parameters establishing the same nine circuit characteristics that govern the phoneme mode. Fig 2 and Fig 3 compare the parameters established in each of these two modes for the word "hello".

Speech Technology also uses synthesis from parameters, encoding extracted parameters at 1000 bps to operate the speech synthesizer. The parameters can be programmed into PROMs for some synthesizer models or (for other models) supplied on punched tape or magnetic tape cassette to be stored in computer memory. The company supplies standard vocabularies with its synthesizers. For instance, the base vocabulary for the Model 200 includes the numbers 0 through 9, letters A through Z, and various arithmetic-oriented words such as "plus", "minus" and "equal", for a total repertoire of 54 words. A number of options supply additional vocabulary as needed by the user.

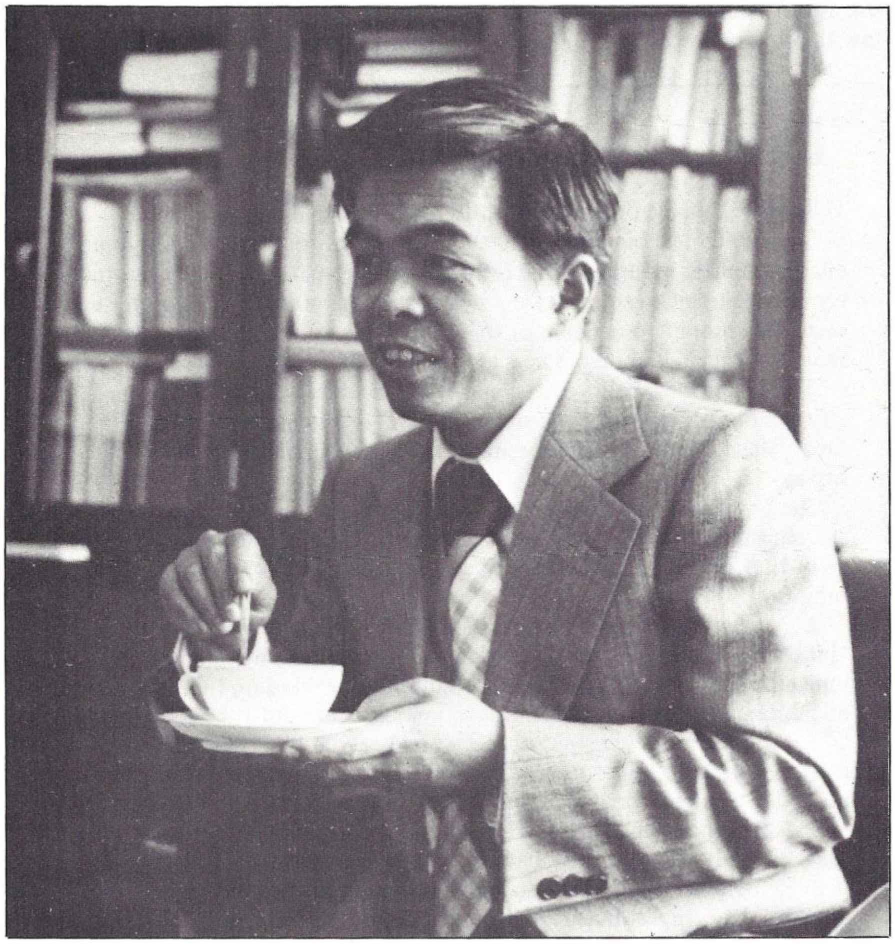
Existing and Potential Use

Applications for talking computers include education, linguistic research, warning systems, monitor systems, aids to the handicapped, training simulations and telephone systems. A talking computer under development by Telesensory Systems in conjunction with M.I.T. exemplifies how much can be done. In the system, an optical reader (able to read print from most standard fonts) will feed into a phoneme synthesizer and result in a spoken output. Thus, this system will read arbitrary printed outputs out loud.

To sum it up, it seems that we can expect computer terminals that can talk to us as well as write to us. In that light, the speech synthesizer programming field will become increasingly active.

PERSONAL COMPUTING IN JAPAN

About a year ago we received a letter from Professor Haruisha Ishida of the University of Tokyo ordering multiple copies of our magazine to be distributed to his computer club in Japan. When Harold Buchbinder was in Tokyo, recently, he visited Professor Ishida to discuss the status of the personal computing industry in Japan. Hiro H. Irie of Personal Computing*'s Tokyo office and Katsuya Araki, of the* Nikkon Kogyo Shimbun *(Japan's largest daily industrial newspaper) accompanied Harold Buchbinder on a tour of the Tokyo University Computer Center and sat in during the following interview with Professor Ishida.*



Professor Haruisha Ishida.

Personal Computing: I've toured your University Computer Center and it appears to be among the largest I've visited. Yet, I noticed that some of your laboratory rooms have mini and microcomputers. Can you tell us why and how your interest spans such a large spectrum of computers?

Professor Haruisha Ishida: I was formerly interested in very large computers but now I am devoting most of my time to small computers. You might say that I have gone from one end of the spectrum to the other. Before January of 1972 I was involved only in the work at the Computer Center and with the big computers they had. In fact, at that time the Computer Center Committee asked me to make the selection of a large computer they wished to buy for the University. I picked the HITAC 8800. It had everything I I wished in a large computer.

P.C.: How large of an operation do you have at the computer center now?

H.I.: We have three HITAC 8800 computers and one HITAC 8700 with a total core memory of 8 megabytes. The system is shared by 4000 research people from 100 other universities. The machine rental is two and a half million dollars and the operating cost is about $800,000 a year. The government bears half the expense. But we cost account every record on the computer. You can see we have a very large computer facility at this one university alone.

P.C.: How did you drift into the lower end of the computer spectrum?

H.I.: Before January, 1972, my friend, Mr. Kamo, now of Intel Japan, showed me an interesting device that I had not seen before – the microcomputer chip 4004. He knew that I was interested in small computers, which is why he showed me the chip in the first place. I agreed with Mr. Kamo – this type of system would be very important in the future. We both wished to pay close attention to this.

P.C.: How did you go then, from the 1972 peek at a chip, into the microcomputer field?
H.I.: The next year – 1973 – Intel made an experimental microcomputer based on the 8008. I brought one of these new microcomputers home from Mr. Kamo's lab and used it with a noisy teletype. I think, therefore, I was one of the very first in Japan to use a microcomputer. Actually, after I saw the first microcomputer chip I wrote a short article on it in a Japanese trade magazine that attracted much attention from different companies.
P.C.: Did you predict in that article what might possibly happen? Would there be a significant market in Japan? An important interest in small computers?
H.I.: I felt that the age of microcomputers was coming. I wrote that in my article. We were going to have a hobby market, I said.

The Latest Index

TOKYO – "Viable or not?" "Will it grow aggressive or defensive?"

To provide reliable answers to these questions concerning embryonic microcomputer industry, MITI (Ministry of International Trade and Industry of Japanese Government) is now questioning 400 microcomputer experts. The looping procedure of their DELPHI method questionnaire should produce definite answers around the end of March 1978.

The Electronics Policy Division of the Machinery and Information Industries Bureau launched the survey for MITI. A management consultant, Japan Management Association, represents the technical aspect of the survey.

Reflecting the rapid growth of the industry, the survey hopes to obtain current microprocessor consumption figures. Official statistics in government files, today, are old and may well be out of pace with actual consumption.

Figures reported by "Shadan Hojin Nihon Denshi Kogyo Kyokai" (Japan Electronic Industry Development Association) in March of 1977, indicated that Japanese domestic consumption, excluding I/O, LSIs, and microcomputer memory LSIs, was estimated at around 1.4 million units. Of these 87.2% are 4 bit microprocessors, 11.3% 8 bit, 1.4% 12-16 bit, and 0.1% bit slice microprocessors. The DELPHI format recipients are asked for more convincing consumption data.

Production potential in developing countries has made production forecasts difficult. The survey polls the experts' views on this. The question is realistic because some Japanese industrial sectors are actually being threatened by the competition from developing countries.

The survey focuses on critical technological and marketing equilibrium between American and Japanese microcomputer industries in the near future. "Is it viable? Would it be wise to concentrate on only parts of the future domestic market? Would it be too large to depend on the entire world-wide market for survival?" The DELPHI method hopefully will answer these questions.

Given the respective shares of locally produced versus imported computers, the survey studies the software gap between American and Japanese makers. The survey further explores the marketing of computers in Japan. The questionnaire also asks recipients their views on the future Japanese market. In particular, the survey speculates on the dominance of the field by American computers and their advanced "operation systems". Out of 6,903 units, costing 614 billion yen, sold during the fiscal year 1975, the imported general purpose computers cut a substantial slice out of those sales (29.4% of the units and 44.2% of the dollar value). American superiority in operation system technology was predominant.

This superiority may well affect the future market of microcomputers in Japan.

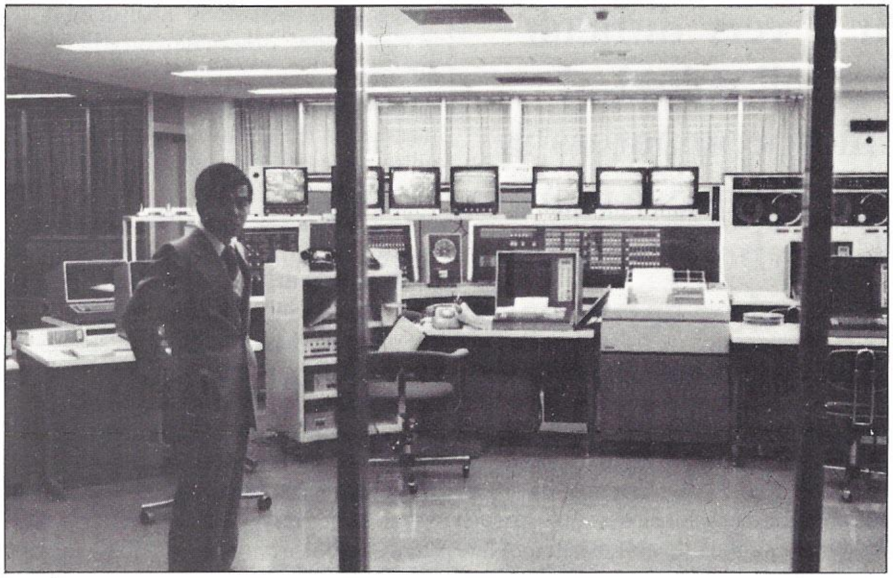
– Katsuya Araki

P.C.: Did you build a hobby computer yourself? Or were you able to buy a computer company's product?
H.I.: I have built computer kits myself. The SDK-80 from Intel was the first one I built. But my interest increased when I returned to the States. In 1975, I was invited by Bell Laboratories to come to New Jersey. I stayed for about 10 months.
P.C.: Working on microcomputers?
H.I.: No. Actually, I was studying computer networks. Although my work was not with microcomputers I was interested in that field. I remember seeing in *Popular Electronics* the ad:
"Let's give microcomputers as Christmas presents this year."
P.C.: Did the ad include a price for the microcomputer?
H.I.: About a thousand dollars. They stressed the idea that it was a Christmas present.

Star Trek games represented the first time microcomputers were used for game playing.

P.C.: What about your interest in microcomputers? How did you sustain that interest?
H.I.: I subscribed to all the microcomputer magazines. I also contacted many clubs in the States.
P.C.: How did you find out what magazines were covering microcomputers?
H.I.: Through the people at the Bell laboratories. They were playing games with computers and I joined them. In fact, I was instrumental in introducing Star Trek games in Japan as a result of this game playing in the States. Until I introduced Star Trek, the Japanese were not playing computer games. Some people were interested in puzzles, but not games.
Katsuya Araki: On what computer did you introduce Star Trek?
H.I.: We put the program on our large computer and spread the word.
K.A.: Was it the same Star Trek game you played on the smaller computer in the States? Or did you enlarge the game? Also, was it more difficult?
H.I.: It was the standard one. Since

then, some of my students have improved the game.
P.C.: With the enlarged memories now available in microcomputers, smaller computers in the States can probably handle a more complicated Star Trek game. When did you get back to Japan?
H.I.: I left Bell Laboratories in January of 1976 and returned to the computer center here at the University. And again my interest grew in microcomputers.
P.C.: Was interest in microcomputers beginning to pop up here in Japan about that time?
H.I.: No, it is almost unbelievable. But until the summer of 1976 we had no microcomputer magazines, no computer stores, no microcomputer kits, and no microcomputer clubs.
Hiro Irie: Who started the first computer club here?
H.I.: I think you could say I started the first club here. It was really called a home-computer club.
Irie: Do you have meetings, now? Do you get together? Swap programs?
H.I.: No. We merely subscribe to American magazines as a way to buy them cheaper. In other words, we subscribe as a group then pass the issues around.
P.C.: How did you recruit your first members for this home-computer club?
H.I.: In January of 1975, I published a book in Japanese on microcomputers and another one in 1977. In each book I discussed American microcomputer magazines and added one line at the end: such magazines, when you join a home-computer club, would be available through the club. And I inserted my address. So people who read the book became possible members and possible subscribers.
P.C.: How did you find the time to form a club? Didn't it interfere with your work here at the center?
H.I.: I started the computer club in Japan because publishers of computer magazines in the States had no real circulation in Japan. One magazine had only five subscribers here. Imagine that, only five in the entire country of Japan! So I decided to pass the word for them here.
P.C.: Did your book on microcomputers have anything to do with creating a demand for microcomputers – the chips?
H.I.: Many people have told me so. But it required a year-and-a-half from

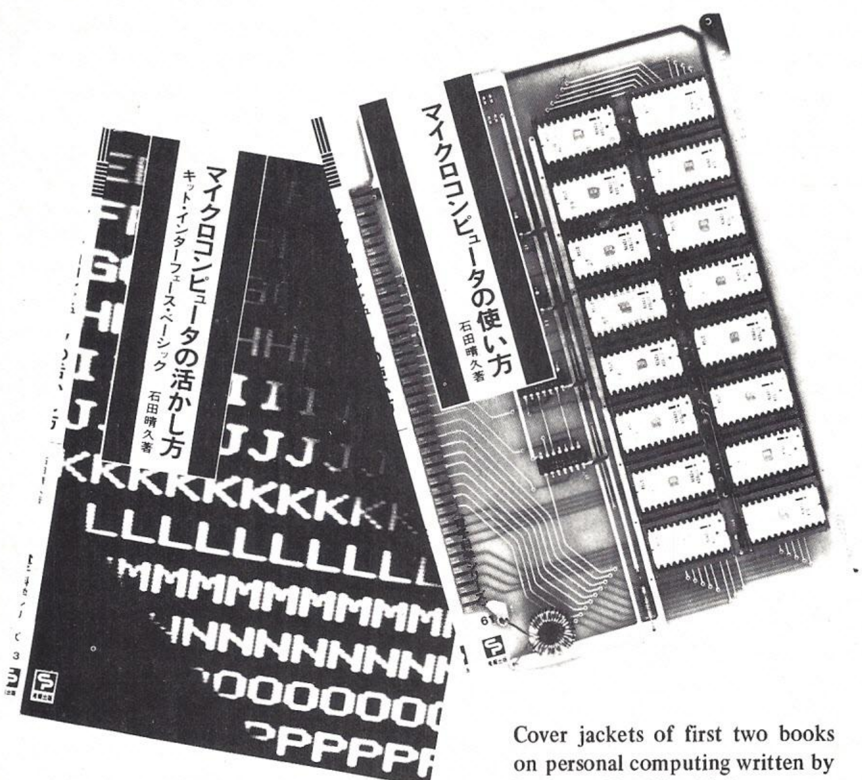


One of the control rooms for large computer in operation at Tokyo University computer center.

the publication of the first book to the opening of the computer market here.
P.C.: Can you tell us something about the current boom in Japan? In other words, what's the status today, what happened from the summer of '76 to now, and where do you think you're going from here?
H.I.: One of the strange characteristics of Japanese business, and especially so in the microcomputer field, is that nobody wants to be the front runner. But once somebody starts to run then everyone jumps on the bandwagon. For example, Nippon Electric introduced a microcomputer kit in the summer of 1976 and this, I would say, started the microcomputer boom. Now we have many kinds of microcomputer kits, microcomputer shops, and magazines of our own.
P.C.: From your point of view, your personal interest, which of these magazines is of greatest use to you?
H.I.: None of them. The articles they carry are very primitive. Very, very low-level. They talk about kit or kit signals or timing or how to cut printed wires, because Japanese kits don't have extended ability. If you want to add memory you have to cut printed wires. And all kits are single-board types with no extended benefits.
P.C.: Are extendable American kits available now?
H.I.: Yes, but they are very expensive. Two or three times as expensive as in the States. That is due to the quantity. Because your kits are so expensive they can't sell many. And if they don't sell many, then the price does not come down.
P.C.: What about the language problem? Japanese who use programs in BASIC must know English.
H.I.: Language is, yes, a very great problem. Many Japanese don't know about BASIC language. Also, because in Japan the time-sharing system has not been used very widely, we have not been using BASIC minicomputers. Our typical minicomputers are very small.
P.C.: What's the memory size?
H.I.: Eight kilobytes. Sixteen kilobytes. They are smaller than today's microcomputers. And we do not have any disk on the minicomputers. That is one of the reasons you cannot use BASIC in Japan.
P.C.: What did you use to store large quantities of data?
H.I.: We have almost no memory storage capacity at all. We use paper tape to load the program. So programming for microcomputers has been mainly on assembly languages.
P.C.: Going back to language problems, for a minute. Do you have anything that is used with Japanese characters? Do you use codes?
H.I.: The only Japanese characters we use are with the KATAKANA keyboard. They are very expensive keyboards to use with microcomputers.
P.C.: How much would the KATAKANA displays cost over American keyboards?
H.I.: About twice as much.
P.C.: That obviously gives the personal-computer hobbyist a real incentive for learning English.

It is very difficult to sell naked microcomputers in the Japanese market. Everybody expects the other services to come along free of charge.

H.I.: We do teach English in our schools. We begin at about the middle-school level, which would be about the 7th grade. In principle, every Japanese should be able to read or speak English. They should be able to handle English with a computer, if nothing else.
P.C.: That should mean that Japanese children going through the school systems, now, represent a good future market for computers. Are they taught enough English to communicate with computers?
H.I.: Incidentally, there is an examination here for college entrance. You have to pass a very strict test to enter. Therefore, if you are interested in a college, you must study English hard in high school, otherwise you will not pass.
P.C.: Where do your current programmers come from? Are they coming from universities or from special schools that teach programming?
H.I.: Entrance into a good university doesn't guarantee a good job. People have a misunderstanding about the university role. And we do have private computer schools where one goes for six months or a year or more. They teach only computer courses.
P.C.: At these schools, is programming taught in English or in KATAKANA?
H.I.: In English. Because, also, they teach other languages such as FORTRAN and COBOL. But as for data – they may use KATAKANA. But the program itself is written in English.
P.C.: Now that computers are increasing in popularity is there a boom in these private schools?
H.I.: No. There was a boom, several years ago. But I believe it's declining now. The schools are not very capable. Many people have been disappointed by them.
Katsuya Araki: Earlier, you were talking about microcomputer clubs. How many do we have in Japan, today?
H.I.: I have one organization here at the university. We call it a microcomputer lab. It is part of our computer center. The purpose of this lab is to teach university faculty and students how to use microcomputers. Especially those involved in experiments. The lab allows them to put microcomputers into their experimental apparatus.
Katsuya Araki: Are the computers at the lab only for the engineering sciences?
H.I.: No. For the most part, the engineering community has its own computers and its own microcomputers. So our programs are used by faculty and students in other sciences like medicine, physics, social sciences.
P.C.: What kinds of problems do you have training these users?
H.I.: These people can handle programs in FORTRAN but they know nothing at all about hardware.
P.C.: Do you teach them machine language?
H.I.: Yes, I do. I also lend them the microcomputers. I have a collection of them. Seventeen microcomputers. Different makes. Not all of them are in the labs now. I think I have ten out on loan. They are mostly of the single board type. Not too big.
P.C.: Your students can come in and borrow a computer any time?
H.I.: If they wish, yes. And if I have one available.
P.C.: Where do they take them?
H.I.: They can take them home or bring them to their own laboratories. But for two weeks only.
P.C.: Do you charge for this service?
H.I.: There is no charge. It is completely free. Many of the microcomputers we have here are donated by various manufactureres. That is why I have so many and why I am able to lend them out.
P.C.: You had mentioned kits before. What do you consider the best type of kit in Japan?
H.I.: The best one is the 68TR, made by Hitachi. However, they are really not kits at all. They come assembled. We really call them training modules.
P.C.: Do you have Japanese-made computers like Altair or Imsai to which you can add floppy disk drives, terminals and printers?
H.I.: Not yet. All are American-made products.
P.C.: Can you buy these American products in Japan?
H.I.: Yes. But they are very expensive. Although when I say there is a microcomputer kit boom in Japan, I mean



Cover jackets of first two books on personal computing written by Professor Haruisha Ishida.

the single board type – $400 is the limit. So I am anxious to see when PET and the TRS-80 are introduced in Japan. Their price is a bit higher but it's going to be a long time before they get here. I understand that there is a production backlog in the States. I hear it will be quite some time before they start exporting.

We have another problem in Japan. Journalists have generally shown great interest in microcomputers. And they carry many articles for lay people on microcomputers. They attempt to stress the eventual, wonderful use of the microcomputer. You can do this thing and that thing with the cheap microcomputers, they say. So people get interested. But when they go to the computer stores, all they can buy is the single board computer. And they have to program it.

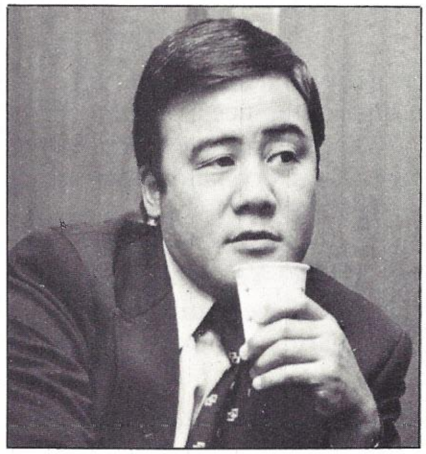
It is quite impossible for the ordinary person to use this kind of microcomputer. The people are being deceived by these articles. They bought the single board computer expecting to be able to do many things with it. Instead, they have been greatly disappointed.

P.C.: Japanese manufacturers in this industry are considered equal and sometimes superior to American manufacturers. Why, then, haven't your own manufacturers come out with the same kind of units now available from so many manufacturers in the States?

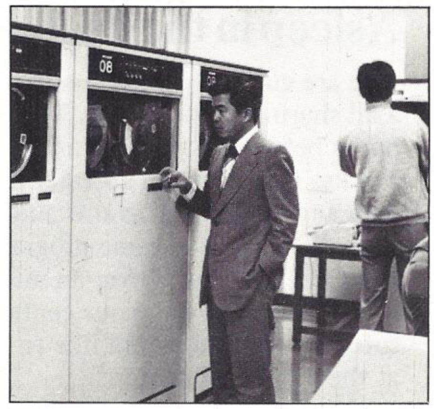
H.I.: There have been microcomputers like Altair in the past but for professional use, only. We have many small companies manufacturing this type of microcomputer. But they're not for the hobbyist. Because they are for professionals, they are built very carefully and they have a very good product. And they have a very high reliability.

We tend to place too much emphasis on reliability in Japan. But, you know reliability is very important. For example, it took me about six months to get my American built computers working. Everytime I got one thing fixed, something else would go. From that viewpoint, and by comparison to Japanese products, American microcomputers are not as reliable.

Let me change the subject for a bit. This publication I have in my hand is called a "mook". It means a combination of a magazine and a book. The purpose of the mook is to interest ordinary people in computers. It contains



Mr. T. Kamo, president of Intel Japan, who brought the first LSI chip into Japan.



Professor Ishida inspects part of the battery of mag tape units at the University of Tokyo's computing center.

Student and faculty credit cards operate various portions of the computer equipment in Japan. Left, Hiro Irie. Center, Professor Haruisha Ishida. Right, Katsuya Araki. In background, two students wait for the computer to finish their printouts.

a map which is very important if you wish to go to a microcomputer shop. These shops are difficult to locate, you see. You have to go to some corner of the city, on a side street, in an old building and climb up to the third floor to a computer store. That is why you need the map.

Say you know which product you want. This is typical Japanese behavior. You make a phone call to the computer shop, rather than visiting it. You order the microcomputer over the phone. But if something happens to the microcomputer when delivered you will have to call the computer shop again to get a repairman to come to your place and fix your machine. Installation, reliability, plus repairs keep microcomputer costs high in this country.

P.C.: What about software?

H.I.: The software, of course, must come free of charge, otherwise you could not afford it. Other costs to be added are delivery, maintenance and repairs. It is very difficult to sell naked microcomputers in the Japanese market. Everybody expects the other services to come along free of charge and when they have to pay for them, they hesitate to buy.

P.C.: You have other books on the desk that look like computer books.

H.I.: These are beginner's books on computers.

P.C.: Are there many of them in Japan?

H.I.: Yes. The Japanese market has many such books for beginners. As I have said, the Japanese are very much interested in the microcomputer and as soon as the price comes down you will see an increase. For the moment, they are content to learn about them

Asleep in the deep.....

"We are currently working on an international network link for load sharing in the future. There is a time difference of nine hours between California and Tokyo. While we are sleeping you can use our chemical abstract data base, X-ray crystallographic data and analysis programs and excellent FORTRAN compiler. We also have game programs such as Musashi, Gomoku and Othello. Games over an international link? Yes, while your boss is out of town. Your centers, of course, have abundant software that we would like to use in January, our busiest month at the computer center."

—Professor Ishida in Perspective, *September 1977*

through books like these.
P.C.: You were editor of a special issue – the Japanese version of the *Scientific American.* Can you tell me something about that?
H.I.: This issue on my desk written in Japanese and by Japanese writers is a special issue on microcomputers. The authors are my friends. Because the Japanese computer market is so small, at this moment, I know almost all the key people in the industry. So I called my friends to help me compile this special computer issue. I had to rely entirely on American products because we cannot as yet get this type of microcomputer in Japan.
P.C.: About how many hobbyists do you think there are in Japan right now?
H.I.: My discussion with manufacturers and computer outlets indicates that we must have about 25,000 kits operational in Japan. All of them are the single board type costing under $400. We estimate the amount of money the hobbyist can afford to spend is around $400.
P.C.: In the United States there are many physicians, teachers, lawyers and other professionals who spend anywhere from $5,000 to $7,000 on a boat – some even more than that. And they only use it four or five months each year – sometimes only on weekends. Many of these people will spend $2,000 or $3,000 for a computer they can use all year. Also, they can now share it with their children. Do you think think there's a Japanese market for computers costing $2,000 to $3,000?
H.I.: Eventually, yes. But in going into that we would have to prepare many programs and that would present many problems.
P.C.: Such as what? Are you speaking now of educational problems, or the problems associated with microcomputer training?
H.I.: First of all, we are not used to the typewriter. We do not type as much as the Americans. So, we do not have many people who like to sit down to keyboards. And it is very difficult for them to use keyboards of any type; Japanese keyboards are very poorly designed and very hard to redesign. Even the national, standard keyboard is very poor. It is divided into four sections. Two shifts for the alphanumeric keyboard and two shifts for KATAKANA. You will have to handle four shifts on the keyboard and that would make the keyboard very difficult to use. Like a big American organ.
P.C.: What about the American keyboard with only two shifts, if users understood English?
H.I.: Of course that would be much better. But there would be some apprehension toward any keyboard at all. We do not teach typing in the schools. So this would represent a new skill and a new training that will have to be learned.

I believe very strongly that we should be teaching typewriting in Japanese schools. I have been advocating this for several years now, but we have another problem to consider. The general use of KATAKANA characters on the Japanese keyboards would be very expensive. So we have the language problem. Time-sharing systems are not widely used in Japan. You don't operate the keyboard yourself. There are people trained just for that. So, all in all, the introduction of microcomputers to Japan would be very useful.
P.C.: Is any effort made to introduce or stimulate interest among high school or college students in computers?
H.I.: Many newspapers in Japan are now reporting on the importance of using computers. Newspapers are a good way for Japanese to become interested in microcomputers. High school kids become interested from articles in newspapers. And, as a result, I have met many high school students who say they want to use microcomputers.
P.C.: Many high schools in the States now have computers available for stu-

The Night The First Chip Landed

In the preface to his book, How to Use Microcomputers, *Professor Haruisha Ishida recalls the night he met the 4004 chip. He found the experience traumatic, as you can tell from this English translation of the Japanese text supplied by Katsuya Araki.*

One evening in January of 1972, a group of us were having dinner at a fancy sky-top restaurant in one of Tokyo's tall buildings. Besides myself, there were my friends, Mr. Kamo of Intel, Japan, and Mr. Kitahara, an electronic distributor. In between our dinner courses we held heated discussions about the computer business in general – it's nice to have heated discussions on cold nights in Japan's winter season.

Looking out the windows of the restaurant and across the rooftops of Tokyo, we could see the fascinating glare of the millions of lights that illuminate our great city. Suddenly, one of the group pointed to a small object on the table. Mr. Kamo had brought it from the U.S. to show us. It was an LSI – a general purpose single chip CPU! This tiny thing that had just been brought to Japan was destined to start this whole microcomputer madness that has engulfed our country.

dent use, so some are familiar with computers before they get to college. Is there anything like this happening in Japanese high schools?

H.I.: No. There is no high school market as yet. You see, many people are wondering what use they can make of the computer, anyway, if they got one. It's not easy for ordinary people to use. Japanese children can't understand English. Even if you try to let them play with games the messages are in English. High school students are not proficient enough in English or typing even with 2 fingers to use computers.

P.C.: Is is possible to make a breakthrough to the computer using Japanese characters? In other words adapting the Japanese language to the computer?

H.I.: That is not likely to come about because the nature of the language almost prohibits it. English has 26 characters. Japanese has 2000 characters, at least.

P.C.: That would make for a horrendous keyboard. Somebody mentioned an organ keyboard. I see what you mean.

H.I.: Yes. Using English when we talk with computers will give students the incentive to learn English if they ever hope to advance to college and to the sciences. So far, we have been teaching only literature in the English language. Our students probably know more Shakespeare, for example, than your students do in the States. It is strange that we teach Shakespeare but we don't teach the more important technical languages. We should be placing more emphasis on technical languages.

P.C.: Those students who aren't too happy with Shakespeare might find the use of technical dialogue a welcome relief. Shakespeare isn't very popular among students in the States. But we do note how many Japanese students try to write in English no matter how poorly they do it. What would letters look like if they were written by Americans using Japanese characters?

H.I.: The computer center here has a problem, too. We require English technical assistance. There is no word-processing market in Japan. One of the areas I am working on at the center is text-editing assistance for English composition because we have to write papers in English to be read by scientists in other countries.

P.C.: If typing is such a problem in Japan, how does a Japanese businessman write a business letter?

H.I.: He does it himself. Sometimes he can dictate the letter to a secretary. Mostly he writes it by hand, himself. Sometimes letters are very vague for that reason. But they are elegantly vague with nice scrolls and flourishes. Official papers, however, are not handwritten. They are typewritten, which is very expensive in Japan, because it must be done by professionals. Also, publishing costs have skyrocketed. That's why publications are very expensive here.

P.C.: Have you thought of using optical character recognition to get rid of the keyboard. You could have routines handwritten, and then pass the optical scanner over it and go right into the computer.

H.I.: About two years ago we did try that technique. We borrowed an optical character reader and tested it for use by ordinary people. But then we ran into difficulties. Many users for the optical system write characters in many different ways. And the machine, we found, could not handle this problem. Also, to make characters familiar enough for the scanner to recognize they would have to be written very carefully in a specified square. It is very tiring to write in a restricted way. Our users did not like this system for that reason and we gave up on the test.

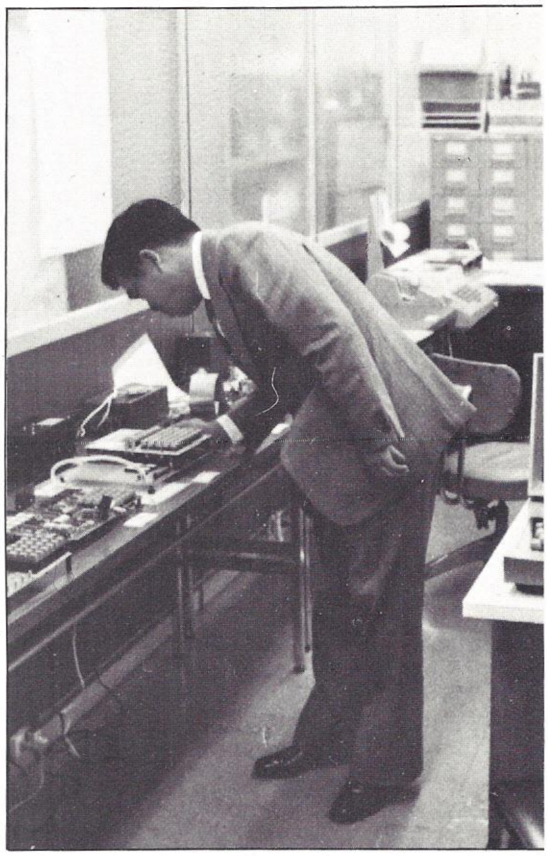
P.C.: Then, you function like an experimental test lab, too?

H.I.: Yes. But they must be confined to computer problems. By the way, that is how we get so many products at half price. We buy the test models and report back to the manufacturer.

P.C.: What other kinds of tests do you run?

H.I.: Many companies bring their terminals here before they start marketing and almost once a week some person will bring in a terminal for testing. We also test displays. And these days, of course, we test the microcomputers. My personal observation is that Japanese manufacturers are very good at mass production because they get the bugs out before they start. If the PET computers and the Heathkit computers sell very good in the States then the Japanese manufacturers would make an all out effort to make such products themselves.

P.C.: Are computer clubs active in Japan? How many members are in your own club?



Part of display of microcomputers available for loan to Professor Ishida's students.

It is strange that we teach Shakespeare but we don't teach the more important technical languages.

H.I.: One hundred sixty. And there are two or three nationwide clubs. When they have a national meeting it is not possible to get them all into one place. One club has 2000 members. The possibilities of Japan becoming a big market for microcomputers is very good. Remember that we have ten million people here. And it would not be too difficult to get tens of thousands interested from one year to the next.

P.C.: What is the most important help you get now from the States.

H.I.: I would say that it is your magazines. They are very helpful to us here and very excellent. They are probably the best help we are getting now from the States. And for that I thank you and ask you to keep it up.

MACHINE LANGUAGE

BY JOHN PALMER

You now have your own personal computer and you think you're doing pretty well learning BASIC, playing Star Trek and figuring your income tax. Then you come across an article that tells you how to do new and wonderful things with your computer if you'll toggle in a short 96 byte machine language program – and plug your electric toaster into your I/O device!

Afraid to take a chance? Plugging in the toaster could only burn the house down. But machine language. That seems too much to ask of a beginner!

If binary code mnemonics (can't even pronounce it) and assembly language strike fear into you, then read on. There is a simple way to try a few lines of machine-level programming without learning a new language. Do it in BASIC!

Most 8K BASICs have some provision for modifying the contents of memory without touching the front panel of your personal computer. In Micro-Soft BASIC you do it with the POKE statement. Look at your user's manual to see how to do this in the BASIC you're using.

Many of the machine language routines published in books and magazines use base 16 (hexadecimal) notation while BASIC requires everything to be in decimal. Therefore, all hexadecimal addresses and instructions must be converted to their decimal form.

Since the BASIC you're using probably does not have a provision for converting numbers for one base to another, look up the numbers in a chart or table. The 6800 programming manual lists all machine codes in hexadecimal, octal and decimal. Also, the 8080 programming manual has an extensive 6-page chart. As well, numerous other charts are available on the market.

Equipped with this information, your first try at machine language might follow these steps: 1) Copy a program in hexadecimal code onto a sheet of ruled paper. 2) Translate each hex number to decimal. 3) Using POKE (or its equal) put the decimal numbers into proper memory locations.

This method will work. But it has some drawbacks – believe me, I know. Converting numbers from one system to another invites human error. Furthermore, once the program is in its decimal form it's hard to recognize what it does. It wasn't until I'd begun to memorize all of the 8080 codes in decimal that I discovered the "beginner's method".

With the beginner's method BASIC does more of the work. The string functions make it possible to write a BASIC subroutine to do the base conversions. Follow these three steps: 1) Copy BASIC program for beginner's method. Save it on tape or disk. 2) Write machine language program using the symbols found in the microprocessor user's manual. Check the meaning of each symbol. Include a few NOP instructions. 3) Convert symbols to hex numbers. Enter hex numbers into data statements of the BASIC program.

Don't try to memorize hex codes for each instruction; learn the symbolic names for the machine codes. As you gain experience, you'll recognize the value of mnemonics in helping you remember what a certain instruction does.

Before trying to run a program, check for errors. Make changes before, not after, running! And don't worry, your machine will *not* self destruct if you place your machine language routine into a memory area that is being used by BASIC. This stops many beginners cold, but, have no fear. You can't destroy your computer by bad programming.

Make a tape or disk copy of a program before you tell the computer to RUN. Then, if a small mistake should "eat up" all the memory simply reload both BASIC and the program and correct the error.

For your memory addresses you'll have to convert a hexadecimal address into two separate decimal numbers. This is not really very hard. Just imagine that your computer's memory is like a book with 256 pages and exactly 256 words to a page. To locate any position in memory, then, all you need to know is the page number and the word number.

You may recall that "K" represents one thousand. (Yet, for some odd reason, it is really 1024 in the computer system. But let's not get side tracked. Just accept it. K=1024.) So then, what is 16K? Ask your computer to print this: 1024 * 16. It should respond 16384.

Here, now lies another problem. For some unknown reason the first memory location is called zero; the second is one; third is two, and so on. The locations referred to as number 16383 is really the 16384th location! Again, don't worry about it. Just accept it.

You ought to reserve about 32 bytes of computer memory for machine language. (BASIC must not use the same area.) When setting the memory size, make it 16352 or less. (This is for a 16K system.) BASIC will now use everything up to and including 16351.

Now what, you may ask, is 16352 in hexadecimal? How do you split it into two numbers, a 'hi-byte' and a 'lo-byte'? Remember the 256 pages in the memory. Simply divide 16352 by 256 and the answer is 63.875. So the page is number 63 in decimal. This will be the 'hi-byte' of the address of the machine language routine. Now exactly where on page 63 does the routine start? Multiply 63 times 256. That gives 16128. Subtracting 16128 from 16352 leaves 224. Logically, then, the routine begins at word 224 on page 63. The 'lo-byte' is 224 in decimal.

To convert to hexadecimal look up 63 and 224 on a conversion chart. In hex, 63=3F. 224=EØ hex. Then, 16352 decimal must be 3FEØ hexadecimal.

When writing out machine language code, start with the hex number 3FE∅ in the upper left hand corner of your sheet of lined paper. Number each line with consecutive hex numbers. (You may also want to put the decimal numbers alongside to help in checking your work later.)

In the BASIC program each DATA statement, except the last, has eight entries to help when checking your BASIC program against the hand-written code. Every item *must* be two characters. Enter a zero as "∅∅". Don't use a comma after the last item. The line numbers indicate the decimal word location of the first item on each line. This is another aid to the beginner.

For your first attempt with beginner's method try my sample program (see Program 1). The BASIC program reads items from the DATA statements and converts each to decimal. The operating system stores data in memory and prints memory location. Next, the memory location increases by one and the process repeats. A value in a DATA statement that is not a legal hexadecimal character will cause the program to go to its final steps. Line 299, value "XX", forces the program to its conclusion. This program works if: 1) You use a computer like the ALTAIR 8800. 2) You use MITS 8K ALTAIR BASIC REV. 4. 3) You rewrite DATA statements and lines 310, 370 and 380.

After running this program and BASIC has returned command, you may erase the program. The machine language routine will still be there. Although some people may argue that this method wastes computer memory it really doesn't. Once the routine enters memory the program can be erased and another program entered. And if you're going to use BASIC as your principal language, this method is quicker than loading an assembler and then reloading BASIC. However, if you want a routine that has more than two or three hundred bytes of code, consider a more advanced method.

What does the model program do? It turns the interrupt light on and off by using the USR function in BASIC. Consult the machine code listing to see how it does this. For another CPU you must rewrite the code. When using an 8080 plus a different version of BASIC, check the language manual to find the correct term for the USR function. You must know where in BASIC to place the address of the machine language routine. Also, you must know how the argument of the USR function passes to the routine.

After copying the sample program and necessary changes run it then erase it. Now type in the second program (see Program 2). This one is much shorter. I call it "musical interrupt". It's not really very musical, but it will help you understand how your personal computer can do novel things with just a little bit of programming effort.

RUN the second program. When it asks for "duration" and "tone value" start with values of 100. Upon hitting the return to the second input, BASIC will call the machine language routine and the interrupt light will glow or flicker. Place an AM radio near the interrupt line. An "almost" musical sound will be heard coming from the radio!

Although not new, (many fellow computer hobbyists have been doing this for some time) if it is *your* first time you'll feel like you've really accomplished something.

What next? Once you have the beginner's method mastered, you're limited only by your imagination. So why not try a little machine language today with the help of good ol' 8K BASIC? It's the beginner's best friend!

Machine language listing

```
224  3FE0   CD   CALL      get routine in BASIC that returns
       E1   38             with argument in registers D & E
       E2   07
       E3   15   DCR D     decrease D by one
       E4   C8   RZ        return if zero (D=0)
       E5   7B   MOV A,E   register A get value in E
       E6   00   NOP
       E7   00   NOP
232    E8   3F   CMC       turn over the carry flag
       E9   DA   JC        jump if the carry flag is up
       EA   ED             (it will jump over the
       EB   3F              EI instruction)
       EC   FB   EI        turn on the interrupt line
       ED   00   NOP
       EE   00   NOP
       EF   00   NOP
240    F0   3C   INR A     increase the value of A register by one
       F1   C2   JNZ       jump if not zero (A = 0)
       F2   E9             (kill some time by doing
       F3   3F              some things over again)
       F4   F3   DI        turn off interrupt
       F5   03   JMP       jump back to the
       F6   E3               DCR D instruction
       F7   3F

                            NOTE: don't confuse the CMC
                            instruction with the 'hi-byte'
                            of the address. Both are 3F hex!
```

Program 1

```
224  DATA   CD, 38,07,15,C8,7B,00,00
232  DATA   3F, DA,ED,3F,FB,00,00,00
240  DATA   3C, C2,E9,3F,F3,C3,E3,3F
299  DATA   XX
300  REM    START AT HEX 3FE0=63*246+224
310  M=63*246+224
320  READ   HS:IF HS  "FF" THEN GOTO 360
330  GOSUB  400
340  PRINT  M,D:POKE M,D
350  M=M+1:GOTO 320
360  REM PUT ADR IN USR
370  POKE  73,224
380  POKE  74,63
390  END
400  XS=RIGHTS (HS,1)
410  GOSUB 450:D=X
420  XS=LEFTS(HS,1)
430  GOSUB 450:D=16*X+D
440  RETURN
450  X=VAL(XS)
460  IF XS="0" THEN RETURN
470  IF X=0 THEN  X=ASC(XS)-55
480  RETURN
490  REM BEGINNER'S MACHINE LANGUAGE. JB PALMER 9DEC77
```

Program 2

```
 10  PRINT "D MUST BE    128, E  256"
 20  INPUT "DURATION D ";D
 30  INPUT "TONE VALUE E";E
 40  U=D*256+E
 50  R=50
 60  FOR N=Z TO R:A=USR (U):NEXT
 70  GOTO 20
 80  REM MUSICAL INTERRUPT
```

The disk system you want at a price you didn't expect from a company that understands systems.

THE VISTA $50 FLOPPY DISCOUNT

We know that one of the biggest problems in personal computing is that you're buying with your own personal dollars.

That's precisely why you're going to like doing business with us.

We're Vista Computer Company, the personal computer systems brainchild of the business computer systems people at Randal Data Systems.

And our V80 Floppy Disk System is a perfect example of how we're prepared to help you get the most out of your personal computing dollars.

$649 buys you the whole kit and kaboodle

The $649 you spend on a Vista V80 Floppy Disk System ($749 assembled) gets you everything you need:

An 80K byte minifloppy drive (assembled and tested) that can be powered directly by your 8080 or Z-80 computer. (Case and power supply optional.)

An I/O cable and a single card, S100 bus-compatible controller kit that handles up to four drives and includes a PROM for bootstrap loading (additional drives just $399).

VOS, the most advanced microcomputer disk operating system available, and our BASIC-E compiler, designed to work with VOS, all on a single diskette. Software functions include instantaneous program loading, named dynamic files, program editing, assembling, debugging, batch processing, and file copying on back-up diskettes.

All backed by the Vista 90-day warranty, membership in VUE (Vista Users' Exchange), and Dataforce, our associated service company with 115 locations throughout the country.

Test drive the V80 at your local computer store

Drop by your nearest computer store and run the V80 through its paces. Once you find out what it can do for you, you'll see that our combination of high performance and low price is hard to beat and easy to take.

We love to take orders

If you'd like us to ship you a Vista V80 Floppy Disk System, they're available now. Just send us a check or money order for the amount of purchase, or your BankAmericard/VISA or Master Charge account number with expiration date and authorized signature. California residents add 6% sales tax. Uncertified checks require six weeks processing.

To place your order, or to obtain further information, call or write today.

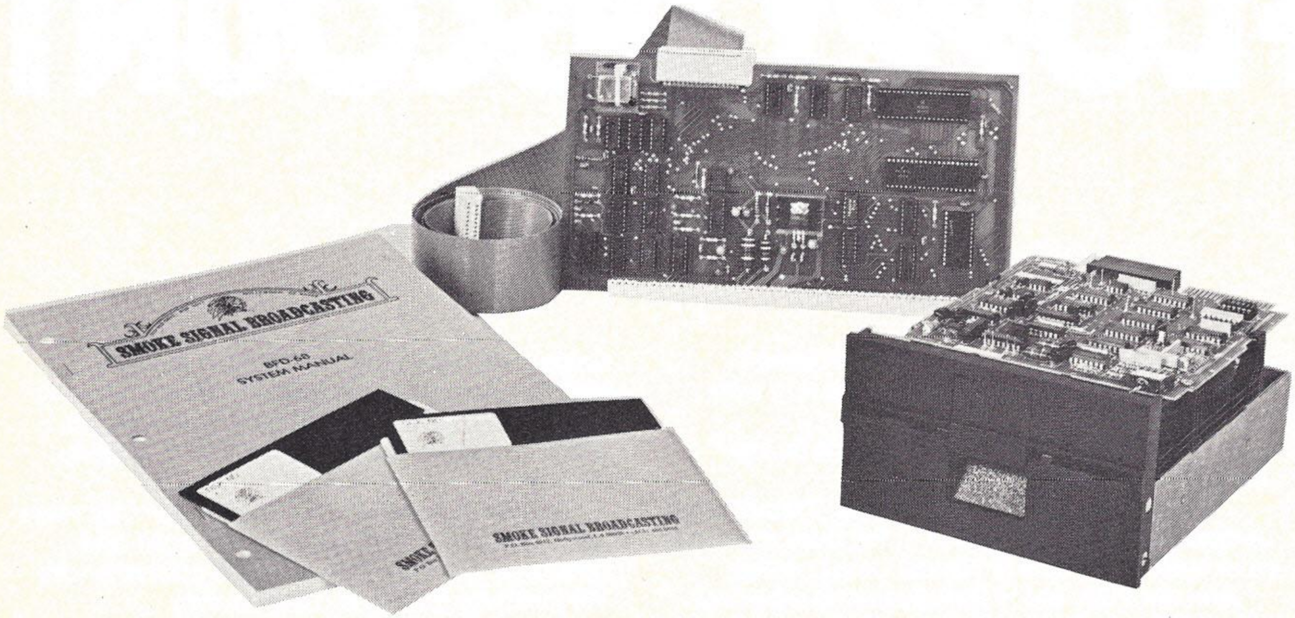
Vista Computer Company, 2807 Oregon Court, Torrance, CA 90503. (213) 320-3880.

Vista

We never forget it's your pocket.

CIRCLE 14

THE SSB $150 FLOPPY DISCOUNT



Affordable

The tribe at Smoke Signal Broadcasting took our BFD-68 disk system and scalped the price, but not the features to create the ABFD-68 (Affordable Basic Floppy Disk). We appreciate the fact that the computer hobbyist gave us our start and we haven't forgotten you.

$649 Assembled

Compare Price. Our SS-50 bus compatible disk system is $150 less than the assembled price of the leading S-100 disk system. And you can at least double that savings when you buy one of the computers manufactured by MSI or SWTPC that use the superior 6800 microprocessor.

Programmable

The BFD-68 is well known for its fine software. The system comes with the best disk operating system available and we offer a multitude of other compatible software products. These include a BASIC interpreter with disk file handling capability. By the way, our DOS now easily handles true random access files as well as sequential. Also, we have a super fast BASIC compiler for business applications. In addition, a Text Editor, 2 Assemblers, a Trace Disassembler useful for program debugging and an Object to Source Code Generator are all stock items available for immediate delivery.
A word processor will be available very soon.

Reliable

We delivered our first mini-floppy disk system a year ago — 6 months ahead of any other 6800 based mini system. Thus, we've had twice the experience in building reliability into the system. Our NEW disk controller was designed using all we have learned in the past year about system reliability.

The ABFD-68 contains all the built in reliability of our regular BFD-68 plus you save money by supplying your own cabinet and power supply for the disk.

Available

We've shipped literally tons of our BFD-68 disk system in the past year and have learned to keep our production up with demand. Give us a call and chances are we'll be able to ship you the new ABFD-68 from stock and charge it to your Master Charge or Visa card. Better yet, ask us for the name of the computer store nearest you that carries our complete line of computer products.

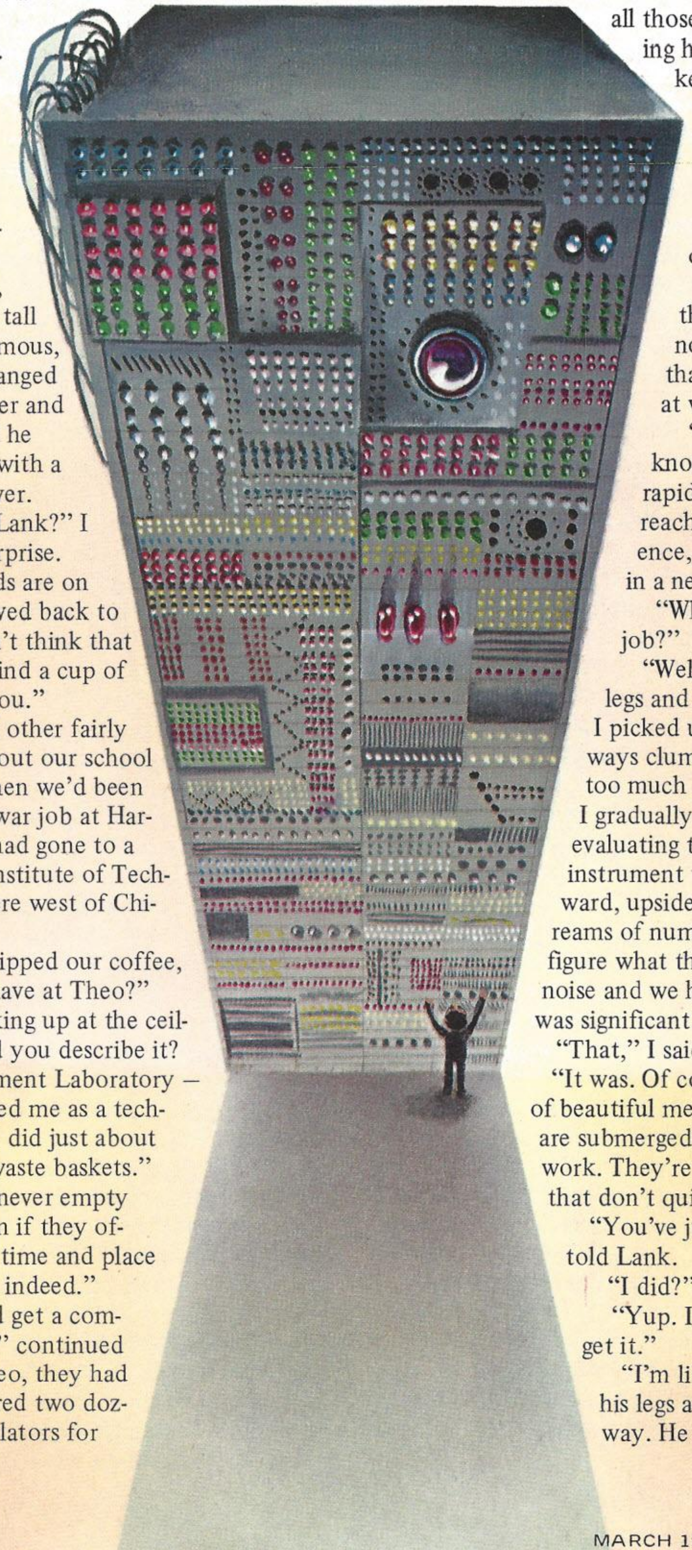
SMOKE SIGNAL BROADCASTING

P.O. Box 2017, Hollywood, CA 90028 • (213) 462-5652

CIRCLE 17

LOOKING BACK

BY HENRY BRAINERD



As I stood on the platform waiting for an overdue subway train, a heavy hand fell on my shoulder. The hand was the size of a catcher's mitt and it gripped my anatomy with the force of a 6-inch iron vise. A voice boomed above my head.

"Hank!"

I turned around and saw my old, dear friend Lanksolot Sniggerson. His once fiery-red hair had turned snowy white, but his craggy features, his kindly but gruff voice, his tall skinny body (6' 6") and enormous, powerful hands were all unchanged from the time we were younger and we used to Indian wrestle and he would knock me off my feet with a swift, sneaky Swedish maneuver. "What brings you back here, Lank?" I asked, recovering from my surprise.

"I retired last year. The kids are on their own, so Karen and I moved back to the old homestead. Say, I don't think that train's ever coming. Let's go find a cup of coffee. Gee, it's good to see you."

From then on we saw each other fairly often. We would reminisce about our school days, the intervening years when we'd been far apart, meeting again on a war job at Harvard. I knew vaguely that he had gone to a job at Theo – the Theobart Institute of Technology – which was somewhere west of Chicago.

"Say, Lank," I said as we sipped our coffee, "what kind of a job did you have at Theo?"

"Let's see," said Lank looking up at the ceiling for a minute. "How would you describe it? It was in their Military Instrument Laboratory – we called it the Mill. They hired me as a technical writer but after a while I did just about everything – even emptying waste baskets."

"Well I tell you, Lank. I'd never empty waste baskets at Harvard. Even if they offered double salary. There's a time and place for everything. Waste baskets, indeed."

"I kept wishing they would get a computer like Aiken's at Harvard," continued Lank. "When I first got to Theo, they had built a gyro gunsight. It required two dozen dames punching desk calculators for a year to evaluate it. They had to compare the way it was set up with what it would need to score hits. My first assignment was to write a report summarizing all those figures; I couldn't help thinking how much faster and better Aiken's machine would have done the job. And, of course, today, you'd write a program, enter a ballistics table and sets of parameters, and, wham! get a stack of printouts and a big roll of graphs in just minutes."

"We've sure moved ahead in all these years. Makes a man feel old, no matter how young he is. Does that make sense to you? The speed at which we're moving forward?"

"Not really," said Lank. "I don't know why everything is moving so rapidly. Whenever you think you've reached an ultimate position in science, bingo! You hit a new jackpot in a new field."

"What happened after that ballistic job?"

"Well," said Lank, crossing his long legs and knocking over a nearby chair. I picked up the chair. Lanky guys are always clumsy and I've learned not to pay too much attention to them. "Somehow, I gradually shifted from writing reports to evaluating tests. Every time we built a new instrument we tested it forward and backward, upside down and inside out. We got reams of numbers and then had to try and figure what they meant. They were mostly noise and we had to dig in to find out what was significant."

"That," I said, "sounds like a helluva mess."

"It was. Of course you can find any number of beautiful methods for recovering signals that are submerged in noise. But try and make 'em work. They're always based on assumptions that don't quite fit your case."

"You've just reminded me of a joke," I told Lank.

"I did?"

"Yup. I gotta tell it to you before I forget it."

"I'm listening," said Lank, unfolding his legs and folding them back the other way. He has a nervous habit and does it

Illustration by David Gardner

constantly. Started doing it when he was in the third grade, he told me, when he was too bashful to ask the teacher to let him go to the bathroom. And he had to go a lot. Never did go into it with him.

"The ark landed," I said, starting my joke. "And Noah told the animals 'Go forth and multiply.' But the snakes stayed behind; 'We can't multiply. We're adders,' they said. So Noah chopped down a tree and built a table. 'Now get along with you,' he said. 'With a log table, even adders can multiply.' "

"That's an old joke," grumbled Lank. "I've heard it three times."

One thing I can't stand is a guy that lets you tell a joke then tells you that he's heard it before. The least he can do is leave the room while you're telling it.

"Well, then tell me more about yourself," I said, maybe a little brusquely. I hate to get brusque with my friends, but sometimes I can't control my temper.

"About that time we got an IBM 650. It would seem crude today but *then* it was a wonderful advance."

"I remember it, Lank," I said. "It was the first real commercially produced computer. They said it had two thousand vacuum tubes and six thousand vacuum diodes. And it took quite a few kilowatts to run it and as many more to air condition the room to cool it."

"Right! And it still had a lot of vestiges of the old IBM punch card systems. Input and output were punched cards. You had to take the output cards to an off-line printer. And what we now call format control was handled by a plugboard."

"You know, Lank, I always thought it should be called a punchboard. It was so much like those gambling gadgets where you pushed a plug of paper through the hole for a nickel and you got either a good luck message or a prize of fifty cents. I once won a quarter."

"The memory in those days," said Lank, "was a magnetic drum with a capacity of two thousand words to hold the program as well as the data. Then, in case you didn't need that much, they had a cheaper model with only a thousand words. I doubt if they sold any of those. It was a decimal machine. The input and output was Hollerith punch code. Storage on the drum was binary coded decimal . . . arithmetic was performed in bi-quinary."

"Bi-quinary?" I interrupted. "That's an old timer. Haven't heard that word since I wore knickers and slickers and had college ices down at the local ice cream spa."

"I'll bet many people today have never heard of that either," said Lank, giving his long legs a quick pretzel turnover. "That quinary part had steps valued 0, 1, 2, 3, 4 and the binary part was 0 or 5. The number 8, for example, was represented by 5 in binary and 3 in quinary. I was always fascinated at the speed and complexity of circuits that would translate one code into the other. Of course, today, we'd say it was simple, just a handful of diodes and gates that a kindergarten kid could put together."

"How big was the old gazooker," I asked Lank.

"A giant," said Lank. He jumped up on the seat of the chair and held his hand in the air as though he were measuring off the height of a field of corn. "The top of it came up to about here, I remember," he said. "I know because I used to keep my lunch bag on top so no one would steal it. Funny how you remember little things like that." He got down from his perch, seated himself again and gave his legs a couple of practice crosses until he found a position that suited him.

"Comfortable?" I asked him.

"Quite."

"Maybe your shoes are too tight, maybe you need a larger size." Lank wears size 14.

"Nothing wrong with my shoes," said Lank stiffly.

"Just trying to be helpful," I said.

"Everything's just fine." Some people don't like to discuss their problems. Lank was one of them.

"You were telling me how big the machine was."

"Yes. It had a cabinet for the card reader and punch, another for the drum, another for the arithmetic unit. Each took as much floor space as a desk. And I showed you how high they were. Then of course, there was the separate printer and an assortment of key punches, verifying card sorters, and what have you. I remember they put a false floor in the room to have a place to run cables in."

"Say, Lank," I said. "Weren't there some faster machines even then?"

"Sure were, Hank. But they were one of a kind. The Whirlwind at M.I.T. and the Eniac at the University of Pennsylvania."

"I remember that baby, all right."

"Whirlwind filled an entire small building at M.I.T. I think it was binary from the start and I remember hearing that its memory was cathode ray tubes with rasters of spots. It was intended for real time simulation. I think it had a short word, mebbe only eight bits. But they aimed at a million operations a second – two operands, add, and store in one microsecond. I think they came purty darn close."

"Karen feeling all right these days?" I asked.

"Oh, just fine. Probably a little arthritis, like most of us. But other than that she feels great."

"Fine. Just occurred to me to ask. I don't want you to think I was impolite by not asking. But go on with that machine thing you were yappin' about. Can't understand why you stop in the middle and change subjects."

"Well if you didn't ask me questions I wouldn't stop."

"I'm sorry. Didn't realize I was the one asking the questions. Sort of fell asleep there for a minute. Talk in my sleep, you know."

"About the Eniac. I remember it used flipflops for memory, two vacuum bits per bit and I have a vague impression that it used binary coded decimals with a word length of six or eight decimal digits. I guess it waren't as fast as Whirlwind but it probably fitted the general range of scientific computations rather better, particularly with higher precision from its longer word."

"I notice you like to use farm language. 'Waren't' – now that's a word only a farmer would use. You into farming?"

"Since I've come back. Have a small place where I keep three or four cows, some chickens, beehive, and patches of corn, carrots, blueberries and strawberries. Keeps me busy."

"It's been so long since I've seen you, guess I've completely forgotten how you used to talk about farming. Said you'd get into it someday. I guess you did finally."

"Yup!"

"Would you like to hear another of my jokes, Lank?"

"No, thanks, Hank. I've got to run along now."

"Why is it that as soon as I start to tell a joke you find you have to run off?"

"The jokes aren't bad," said Lank. "It's the way you tell them." He got up off his chair and darted out of the coffee shop, leaving me to pick up the tab.

I intend to catch up with Lank the next week or so to get some more information on those machines. Also to get a bit more dope on all those things he's growing on the farm. I love fresh strawberries.

MEASURING REACTION TIME

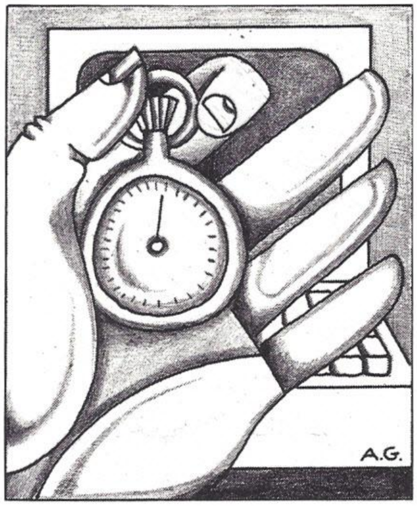
with your Programmable Calculator

BY JAMES C. PITTMAN JR.

Illusions demonstrate an apparent failure in the brain's interpretation of stimuli. Actually, ambiguity of the stimuli and the brain's attempt to perceive what "makes sense" in terms of previous experience produces the failure. Illusions can occur in any sense; the Muller-Lyer double arrow causes us to perceive two lines of equal length as differing in size; ventriloquists produce the illusion that sounds are coming from a source other than the actual speaker.

As strongly visual creatures, humans tend to ignore the contributions of the vestibular balance, kinesthesis (joint and limb location), touch, hearing, and even smell senses to spatial orientation in the environment as long as the sensory inputs agree with each other. We become disoriented when sensory inputs do not agree. For example, many people feel queasy watching a chase scene in a movie. Conflicting sensory information produces this disorientation: touch, kinesthetic and vestibular senses tell the brain that you're sitting still, while your visual and auditory senses say you're dashing around curves and up and down hills. Normally these senses agree; when they don't we feel upset. There are ways to make ourselves more aware of the separate sensory modalities that comprise our everyday sensations. Just turn off the sound on your television and watch the picture for awhile – it won't take long to realize what a large part sound plays in the total information content. Or, turn off the brightness control and listen to the sound alone to see how much information comes from the picture. This rather oversimplified demonstration points out how our perception of events differs when the normal sensory modalities are modified.

At a more basic level of information processing there are other differences in visual and auditory preception. To look at one example use a small portable radio, a programmable calculator with indirect addressing capability, and a program which computes a sequence of random numbers and uses each number to control the duration of a variable-time display.



When the display ends, the program automatically goes to a "counting routine" which serves as your measure of reaction time. The challenge involves stopping the calculator as soon as the variable-time display ends and before the counting routine begins. The faster your reaction time, the shorter the sequence of digits accumulated by the counting routine. The user's visual cue is the normal calculator display of light-emitting-diode segments. The portable radio, picking up the calculator's radio-frequency interference "noise", provides the auditory cue. So the user's reaction time may be influenced by a visual signal, an auditory signal, (something like Pavlov's dogs) or both together.

The program was designed for the HP-67 and HP-97 calculators, but can easily be adapted for use with any calculator with indirect addressing capability. Home computer owners can write their own programs, although some sort of peripheral devices will be required to provide the visual and auditory stimuli as a reaction time on/off switch.

Key the program steps into the calculator and copy the program onto a blank magnetic card for future use. In RUN mode, key a "seed" or starting number (by pressing "f a") as a basis for the sequence of random numbers to be generated. To obtain a sequence of random numbers simply press "A" continually. Program "A" computes two-digit random numbers between 0.0 and 9.9. To start the reaction-time program, press "B". Program "B" will cycle continuously until you press a stop key.

While program "B" is cycling, turn on the portable radio and put it near the calculator display. Tune away from any radio station to get the strongest interference signal from the calculator (different parts of the broadcast band will be more effective than others, depending on the quality of the radio set). Once you have a stable audio signal, watch the program cycle several times to become familiar with the various parts to the visual program in the display and audio heard over the radio.

Pressing "B" also initiates the random-number generator (Program A) to compute and store a new random number. This number is translated to a cue number which flashes on for one second to indicate to the user how long the delay time will be. The program then branches to a series of "do-nothing" steps which provide a characteristic, rapidly-blinking display. At the end of this sequence the digits 1, 2, 3, 4, 5, 6, 7, 8, 9 are entered, one at a time, at a rate of about 70 milliseconds per digit. The change in the display when this sequence starts is the user's cue to stop the calculator by pressing the "R/S" key. A fast reaction time will result in a short sequence of digits such as "1234" or "12345" while a longer reaction time will result in a longer sequence or in missing the digit entry altogether. The user can note or write down the final (largest) digit, then

Sample Reaction Times

VISUAL + AUDIO + NUMBER		VISUAL & NUMBER		AUDIO ONLY	
1234	4	1234567	7	123456	6
1234	4	123456	6	12345	5
1234	4	1234567	7	12345	5
1234	4	1234567	7	123456	6
12345	5	123456	6	12345	5
1234	4	12345	5	12345	5
1234	4	123456	6	12345	5
1234	4	1234567	7	12345	5
12345	5	123456	6	12345	5
12345	5	12345	5	12345	5
mean	4.30		6.20		5.20
Std. Dev.	0.48		0.79		0.42

User Instructions

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Load side 1 and side 2.			
2	Enter a seed: any number.	Seed	f a	
3	Optional: Obtain two-digit random numbers.		A A A	RAN # RAN # RAN #
4	Put portable AM radio near calculator. Tune it away from a station so as to get the best RF-interference sound from the calculator.			
5	Start Reaction Time program. Display sequence: Computation of random number. Display of cue number. Timed display. Counting display. End: 123456789. Program automatically re-cycles.		B	
6	Obtain reaction times: As soon as the timed display changes to the counting display, press R/S. See the string of digits interrupted by stopping the program. Continue by pressing R/S again. Repeat until the program sequence becomes clear. For a fast reaction time, there will be fewer digits in the interrupted display.		R/S	12345. . . .
7	Obtain an automatic reaction time score for ten trials at a time: Press E instead of R/S to stop and restart the program. Now the trial number will be displayed after restart. After the tenth trial, your mean score and standard deviation of your scores will be displayed.			
8	For another set of trials: Press B or E.			
9	Try the program with the radio turned off.			
10	Try the program without the cue number by covering the left part of the display with a card.			

re-start the program by keying "R/S" again. The unused digits are displayed before the program automatically returns to the beginning.

Practice this sequence several times to become familiar with the characteristics of the visual display and the radio sounds. When you're sure you know how all the parts of the program work, you're ready to determine whether your reaction times will differ under different stimulus conditions.

For a simple experiment, turn off the radio and run ten reaction time trials using only the calculator's visual display. Record your scores, the mean, and the standard deviation. Now turn on the radio and run ten trials while keeping the visual display covered with a card or piece of paper. Record the scores again. Now use both the radio sound and the visual display for ten more trials. Plot the data on a table and take a look. Compare your table with the table of Sample Reaction Times. Are they in agreement with what you would have predicted after practicing with the radio sounds and the visual cues?

Repeat the time trials three times to check your consistency and to measure the effect practice has on the results. Challenge others to beat your reaction times. Will a teenager consistently get better reaction times than Mom or Dad? (Probably yes.) Will a male get better reaction times than a female of the same age? (Probably not.)

Variations of this program can be implemented by modifying the cue conditions. To eliminate the cue number before the variable-time display, delete step 044 and replace it with "CLx", so that the cue will simply be a zero for all trials. To change the variable-time display appearance, delete step 047 and replace it with "9", so that the display will be a line of

zeroes instead of mostly decimal points. (The auditory signal will not change much.) Systematically run through your three experimental conditions to see if these cue changes affect your results.

If you're interested in the actual rather than the relative reaction times, you may try timing the digit-entry part of the program. This can be done by keying (in PROGRAM mode) ten digits, then ENTER, ten more digits, ENTER again, and so on until you have 100 digits with an enter after each set of ten. Using a different label, key 100 ENTER steps (ten will run in too short a time) to find the time required by the ten digits separating ENTER steps. Subtract that result from the time required for the whole digit-entry program to run. Switch to RUN mode and time the two programs. Typical values for 100 digits separated by ten ENTERs should be near 7.54 seconds, while for 100 ENTERs you should get about 3.6 seconds. This gives 0.36 seconds for ten ENTERs and therefore about 71.8 milliseconds (or 0.07 seconds) per each digit.

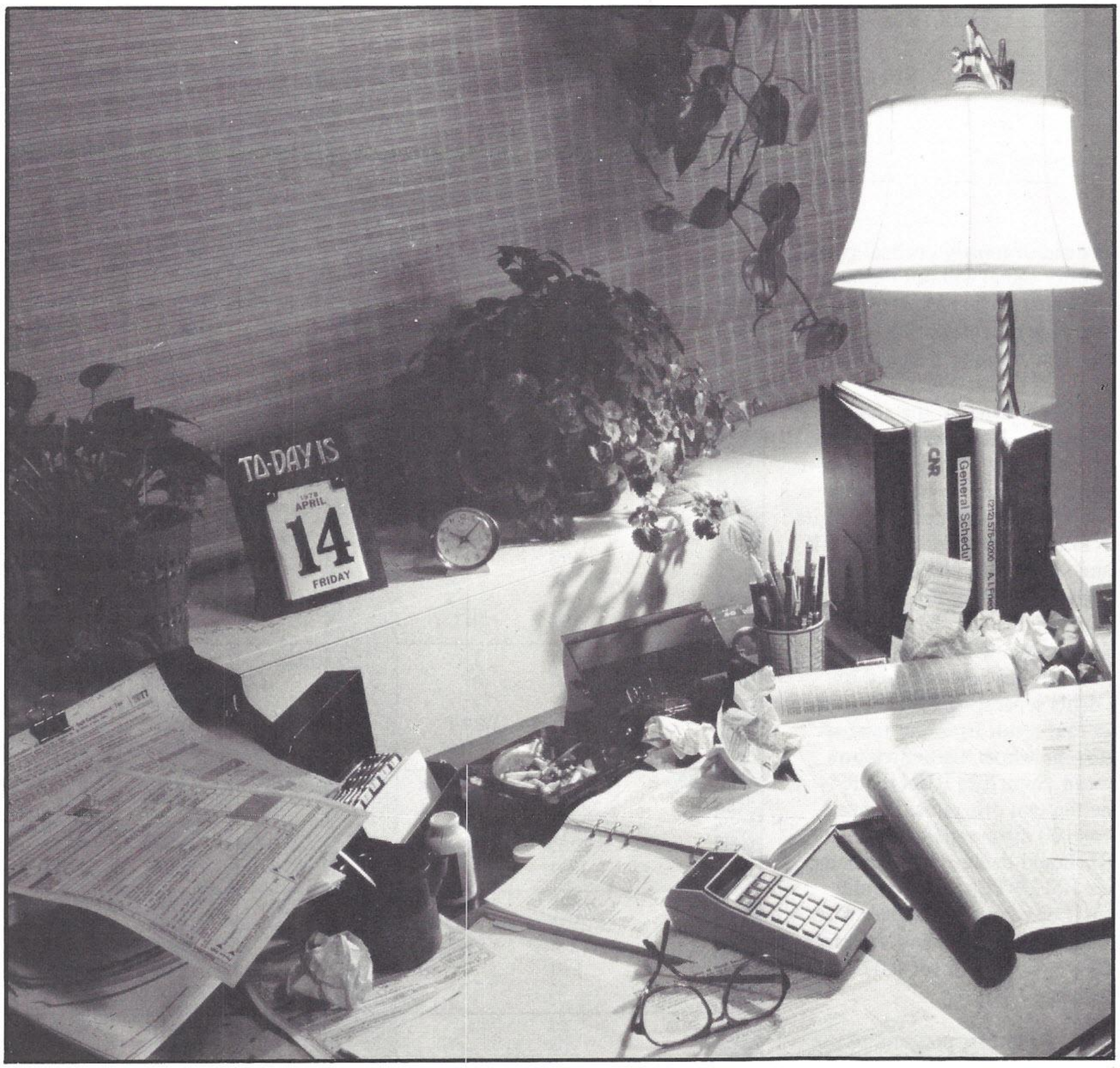
This is not the first program to use interference noise from a portable radio for a practical purpose, nor is it likely to make reaction-time apparatus in psychology laboratories obsolete. It's simply a way for the owner of a personal computing device to experiment with reaction times and their relation to various combinations of visual and auditory cues.

A few interesting variations on the theme might include measuring your reaction time after a double martini or after your first cup of coffee in the morning. Or, perhaps, comparing your alertness and reaction time at 9:00 a.m. and then later at 12 noon. Maybe even checking your reaction time against your biorhythms.

Reaction Time

LINE	CODE	KEYS
001	312511	f LBL A
	357103	h F? 3
	3301	STO 1
	3401	RCL 1
005	3573	h π
	61	+
	05	5
	3563	h y^x
	3238	g FRAC
010	3301	STO1
	43	EEX
	02	2
	71	X
	3183	f INT
015	01	1
	00	0
	81	÷
	2301	DSP 1
	3522	h RTN
020	322511	g LBLf a
	3564	h ABS
	3154	f x
	3283	g FRAC
	3301	STO 1
025	3522	h RTN
	312512	f LBL B
	312211	f GSB A
	01	1
	03	3
030	83	
	04	4
	71	X
	06	6
	01	1
035	61	+
	42	CHS
	3533	h ST I
	06	6
	01	1
040	61	+
	02	2
	08	8
	81	÷
	3124	f RND
045	3572	H PAUSE
	2300	DSP 0
	03	3
	3562	h 1/x
	2224	GTO (i)
050	312515	f LBL E

LINE	CODE	KEYS
051	357102	h F? 2
	2201	GTO 1
	2212	GTO B
	312501	f LBL 1
055	01	1
	00	0
	81	÷
	3283	g FRAC
	01	1
060	00	0
	71	X
	21	Σ+
	3572	h PAUSE
	01	1
065	00	0
	3281	g x>y ?
	2212	GTO B
	3221	g s
	3302	STO 2
070	3121	f $\bar{x}$
	3142	f P≷S
	3143	f CL REG
	3142	f P≷S
	2302	DSP 2
075	3184	f -x-
	3402	RCL 2
	3184	f -x-
	3552	h x≷y
	3522	h RTN
080	3564	h ABS
	3564	h
	3564	h ABS
	3564	h ABS
	3564	h ABS
	3564	h ABS
210	3564	h ABS
	3564	h ABS
	355102	h SF 2
	01	1
	02	2
215	03	3
	04	4
	05	5
	06	6
	07	7
220	08	8
	09	9
	356102	h CF 2
	3572	h PAUSE
224	2212	GTO B



Got the 1040 blues? Liven up your April evenings by letting your computer fill out your tax forms.

Figure 1 - Sample Inputs

```
RUN

TAX RATE FILE ? M-3
DATA FILE ? 1977
 A NUMBER FROM 1 TO 5 AS FILING STATUS ? 2
0 FOR A 24 LINE VIDEO OR 1 FOR THE PRINTER ? 1


ID# ACTION
=== ===================
  1 FORM 1040
  2 SCHEDULE A
  3 TAX
  4 TAX IF PENNIES ARE ROUNDED
  5 PRINT BLANK 1040
  6 BLANK SCHEDULE A
ID#, 1 TO INPUT OR 0 TO REVIEW ? 5,0


ID# ACTION
=== ===================
  1 FORM 1040
  2 SCHEDULE A
  3 TAX
  4 TAX IF PENNIES ARE ROUNDED
  5 PRINT BLANK 1040
  6 BLANK SCHEDULE A
ID#, 1 TO INPUT OR 0 TO REVIEW ? 0,0
FILE TO SAVE DATA ? 1977
 8362
READY
```

DEEP IN THE HEART OF TAXES

A Tax Program for Personal Computers

BY JOE ROEHRIG

When listing usages of personal computers, tax preparation usually finds a place near the top of the list. However, I couldn't find a program that accomplishes this relatively simple task, so I decided to write one myself.

The hardware configuration for the program includes a video terminal (80 characters by 24 lines works best), an 8080 processor and 24K of memory (less if you break the main program into sections). Helpful options include a disk system (I used a North Star Micro Disk) and printer. I've avoided pre-printed income tax forms because they tend to make the application too business oriented. Besides, you would spend at least $50 per box for each of the numerous forms you computerize. However, the programs can be converted to print in the fields of the pre-printed forms.

The three programs detailed in this article calculate, print and store data related to the preparation of Form 1040 United States Individual 1977 Income Tax Return and Schedule A.

The first program creates a blank data file, the second serves as a tax rate file, while the third and main program prepares the 1040 Form and Schedule A. The first two programs are short and require the use of the optional disk system – but you can work around them if necessary.

Figure 1 details the basic inputs you must supply. First you must supply a tax rate file–"M-3" represents my file for married with three dependents. "DATA FILE?" solicits the data file for reading previously supplied or blank tax information. The filing status refers to the five filing conditions appearing in an actual Form 1040. The sample shows a "2" being input representing married joint return filing. An input of "0" for the fourth question makes the computer print only twenty-four lines at a time, a necessity for readable output when using a video terminal running at a high baud rate (say 19,200).

Next, the program prints six actions that you may choose from:

1) View or input to Form 1040. 2) View or input to Schedule A. 3) Print tax owed based on data supplied. 4) The same as 3, but all pennies rounded to the nearest whole dollar. 5) Print a blank 1040 form. 6) Print a blank Schedule A.

After this appears, enter two numeric characters separated by a comma.

Figure 2 - Blank Form 1040

```
INCOME
 8. WAGES FROM W2 FORM                                             8          8.00
 9  INTEREST INCOME (IF OVER $400 ATTACH SCHED B)                  9          9.00
 1A DIVINDENDS        1.00  2A LESS EXCL.        2.00 BALANCE      10C       10.00
11  STATE AND LOCAL TAX REFUNDS                                    11        11.00
12  ALIMONY RECEIVED                                               12        12.00
13  BUSINESS INCOME/(LOSS) (ATTACH SCHEDULE C)                     13        13.00
14  CAPITAL GAIN/(LOSS) (ATTACH SCHEDULE D)                        14        14.00
15  50% OF CAPITAL GAIN DISTRIBUTIONS                              15        15.00
16  NET GAIN OR (LOSS) FROM SUPPL. SCHED.(ATTACH FORM 4797)        16        16.00
17  FULLY TAXABLE PENSIONS AND ANNUNITIES NOT ON SCHED E           17        17.00
18  PENSIONS, ANNUNITIES, RENT, ETC. (ATTACH SCHED E)              18        18.00
19 FARM INCOME OR (LOSS) (ATTACH SCHEDULE F)                       19        19.00
20  OTHER                                                          20        20.00
21  TOTAL INCOME                                                   21        21.00

ADJUSTMENTS TO INCOME
22  MOVING EXPENSE (ATTACH FORM 3903)                  22       22.00
23  EMPLOYEE BUS EXP (ATTACH FORM 2106)                23       23.00
24  RETIREMENT ARR. (ATTACH FORM 5329)                 24       24.00
25  PAYMENTS TO KEOGH                                  25       25.00
26  FORFEITED INTEREST PENALTY                         26       26.00
27  ALIMONY PAID                                       27       27.00
28  TOTAL ADJUSTMENTS                                              28        28.00
29  SUBTRACT LINE 28 FROM LINE 21                                  29        29.00
30  SICK PAY (ATTACH FORM 2440)                                    30        30.00
31  ADJUSTED GROSS INCOME                                          31        31.00


TAX COMPUTATION
32  AMOUNT FROM LINE 31                                            32        32.00
```

Figure 2 - continued

```
33  ITEMIZED EXCESS (SCHEDULE A LINE 41)                         33      33.00
34  TAX TABLE INCOME                                             34      34.00
35  TAX                                                          35      35.00
36  ADDITIONAL TAXES FROM FORM 4970 FORM 4972 FORM 5544
    FORM 5405 SECTION 72(M)(5) PENALTY TAX                       36      36.00
37  TOTAL                                                        37      37.00

CREDITS
38  CONTRIBUTIONS TO CANDIDATES                      38   38.00
39  EDERLY (ATTACH SCHEDULE R&RP)                    39   39.00
40  CHIL AND DEPENDENT CARE                          40   40.00
41  INVESTMENT (ATTACH FORM 3468)                    41   41.00
42  FOREIGN (ATTACH FORM 1116)                       42   42.00
43  WORK INCENTIVE (ATTACH FORM 4874)                43   43.00
44  NEW JOB (ATTACH FORM 5884)                       44   44.00
45  SEE INSTRUCTIONS                                 45   45.00
46  TOTAL CREDITS                                                46      46.00
47  BALANCE                                                      47      47.00

OTHER TAXES
48  SELF EMPLOYMENT (ATTACH SCHEDULE SE)                         48      48.00
49  MINIMUM (ATTACH FORM 4625)                                   49      49.00
50  TAX FROM PRIOR YEAR INV CREDIT (FORM 4255)                   50      50.00
51  FICA ON TIPS (FORM 4137)                                     51      51.00
52  UNCOLLECTED FICA                                             52      52.00
53  TAX ON RETIREMENT (FORM 5329)                                53      53.00
54  TOTAL TAX                                                    54      54.00
PAYMENTS
55  TOTAL FED TAX WITHHELD                           55   55.00
56  ESTIMATED TAX PAYMENTS                           56   56.00
57  EARNED INCOME CREDIT                             57   57.00
58  AMOUNT PAID WITH FORM 4868                       58   58.00
59  EXCESS FICIA                                     59   59.00
60  CREDIT FOR FEDERAL TAX ON FUEL (FORM 4136)       60   60.00
61  CREDIT FOR REGULATED INVEST. (FORM 2439)         61   61.00
 3A SPECIAL                                          3A    3.00
62  TOTAL PAYMENTS                                               62      62.00

REFUND OR DUE
63  OVERPAID                                         63   63.00
64  REFUND                                           64   63.00
65  CREDIT TO 1978                                   65   65.00
66  BALANCE DUE                                                  66      66.00
```

Figure 3 - Blank Schedule A

```
SCHEDULE A
MEDICAL & DENTAL EXP                   CONTRIBUTIONS
 1  INS PREM                   1.00    21  CASH WITH EVID               21.00
 2  MEDICINE                   2.00    211 OTHER CASH                   50.00
 3  1% OF GROSS                3.00    212 ----------                   51.00
 4  LINE 2-LINE3               4.00    213 ----------                   52.00
 5  BALANCE OF INS.            5.00    214 ----------                   53.00
 6  OTHER MEDICAL              6.00    22  OTHER THAN CASH              22.00
 61 DOCTORS,ETC               42.00    23  CARRYOVER                    23.00
 62 HOSPITALS                 43.00    24  TOTAL CONTRIBUTIONS          24.00
 63 OTHER                     44.00    CASUALTY
 7  TOTAL                      7.00    25  LOSS BEFORE INS              25.00
 8  3% OF GROSS                8.00    26  INSURANCE                    26.00
 9  LINE7-LINE8                9.00    27  LINE25-LINE26                27.00
10  TOTAL                     10.00    28  $100 MIN LOSS                28.00
TAXES                                  29  TOTAL                        29.00
11  STATE AND LOCAL INC       11.00    MISCELLANEOUS
12  REAL ESTATE               12.00    30  UNION DUES                   30.00
13  GASOLINE TAX              13.00    31  OTHER                        31.00
14  SALES TAX                 14.00    311 ----------                   54.00
15  PERSONAL PROPERTY         15.00    32  TOTAL                        32.00
16  OTHER                     16.00    SUMMARY
161 ----------                45.00    33  MEDICAL                      33.00
162 ----------                45.00    34  TAXES                        34.00
17  TOTAL                     17.00    35  INTEREST                     35.00
INTEREST                               36  CONTRIBUTIONS                36.00
18  MORTGAGE                  18.00    37  CASUALTY                     37.00
19  OTHER                     19.00    38  MISCELLANEOUS                38.00
191 ----------                47.00    39  TOTAL DEDUCTIONS             39.00
192 ----------                48.00    40  STANDARD DEDUCTION           40.00
193 ----------                49.00    41  EXCESS DEDUCTION             41.00
20  TOTAL                     20.00
```

Figure 4 - Sample Input

```
ID# ACTION
=== ===================
  1 FORM 1040
  2 SCHEDULE A
  3 TAX
  4 TAX IF PENNIES ARE ROUNDED
  5 PRINT BLANK 1040
  6 BLANK SCHEDULE A
ID#, 1 TO INPUT OR 0 TO REVIEW ? 1,1
```

Preprinted tax for
the application too bus
it will cost at least $
of the forms y

The first digit represents the desired action and the second distinguishes between viewing or inputting for items 1 and 2. A "1" stand for inputting and "0" represents just looking. A first digit of "0" ends the program.

When the program ends, you can determine if you want the input of that particular run to be saved. A carriage return ends the execution without saving any new data, while a file name indicates your intention to save the input of the run. (Note: the file name can differ from the name given at the start of the program. This enables you to store different versions of the same tax return.)

The first action was a printout of a blank 1040 (see Figure 2) and the second action ended the execution. The "8362" represents the free memory space. The program is about 11,394 bits long and North Star Software occupies 13,012 bits making the total requirement 24,406 bits.

The blank Form 1040 functions both as an input guide and as a guide for debugging the program (you will probably have bugs since it will be difficult to enter all 241 lines of the program without making an error).

The numbers appearing on the blank without ".00" represent actual line numbers as shown on Form 1040. These numbers are used for all inputs. For example, to input "Wages From W2 Form" 8 is the key index as 20 is the key for "Other".

Numbers with the ".00" indicate variable indexes.

Figure 3 shows a sample blank Schedule A. The numbers represent the same things as those on the Form 1040. One added feature is the "--------". These represent fields where you may

s tend to make
ess orientated. Besides,
) per box for each
ı computerize.

input descriptive data.

During input procedures, these exhibits serve as guide line numbers. If you don't have a printer to generate these guides, you can always use blank Form 1040s and Schedule As. The line numbers (used as input keys) are almost identical to the actual line numbers appearing on the official United States Government forms. The only exception are:

Form 1040
Schedule A

Actual	Per Program
10a, 10b, 10c	1, 2, 10
6a, 6b, 6c	61, 62, 63
blanks-19	191, 192, 193
21a, 21b	211, 212, 213, 214
blanks-31	311

These exceptions were made to simplify the input. I avoided alphanumeric input.

Figure 4 illustrates actual input. Here the user enters the following for his Form 1040:

$20,000	wages
300	dividends
200	exclusion
300	local tax refund
4,200	Federal withholdings

These were accomplished by entering a key or line number followed by a 0 or 1. The "0" makes the current entry override all previous inputs. The "1" adds current input to the existing amount for the particular line number. The third number represents the amount; a "0" line number terminates the input.

Follow the same procedures for entries to Schedule A. Here the computer

Figure 4 - continued

```
ENTER 0,0,0 TO RETURN.  OTHER ENTRIES ARE FORMATTED 'A,B,C'
WHERE A=LINE NUMBER OF ENTRY
      B= 0 TO OVERRIDE PRIOR ENTRY OR 1 TO BE ADDED TO PRIOR
      C= THE AMOUNT
?8,0,20000
?1,0,300
?2,0,200
?11,0,300
?55,0,4200
?0,0,0


ID# ACTION
=== ===================
  1 FORM 1040
  2 SCHEDULE A
  3 TAX
  4 TAX IF PENNIES ARE ROUNDED
  5 PRINT BLANK 1040
  6 BLANK SCHEDULE A
ID#, 1 TO INPUT OR 0 TO REVIEW ? 2,1

ENTER 0,0,0 TO RETURN.  OTHER ENTRIES ARE FORMATTED 'A,B,C'
WHERE A=LINE NUMBER OF ENTRY
      B= 0 TO OVERRIDE PRIOR ENTRY OR 1 TO BE ADDED TO PRIOR
      C= THE AMOUNT
?150.55,0,0
ILLEGAL ENTRY
?1,0,150.55
?11,0,1600.55
?12,0,2400.55
?13,0,300.55
?14,0,1000.55
?161,0,55
DESCRIPTION ? EXCISE
?21,0,100
?212,0,52
DESCRIPTION ? UNITED F.
?25,0,200
?0,0,0
```

Figure 5 - Completed Form 1040

```
INCOME
 8  WAGES FROM W2 FORM                                          8     20000.00
 9  INTEREST INCOME (IF OVER $400 ATTACH SCHED B)               9          .00
 1A DIVINDENDS    300.00  2A LESS EXCL.    200.00 BALANCE      10C      100.00
11  STATE AND LOCAL TAX REFUNDS                                11       300.00
12  ALIMONY RECEIVED                                           12          .00
13  BUSINESS INCOME/(LOSS) (ATTACH SCHEDULE C)                 13          .00
14  CAPITAL GAIN/(LOSS) (ATTACH SCHEDULE D)                    14          .00
15  50% OF CAPITAL GAIN DISTRIBUTIONS                          15          .00
16  NET GAIN OR (LOSS) FROM SUPPL. SCHED.(ATTACH FORM 4797)    16          .00
17  FULLY TAXABLE PENSIONS AND ANNUNITIES NOT ON SCHED E       17          .00
18  PENSIONS, ANNUNITIES, RENT, ETC. (ATTACH SCHED E)          18          .00
19 FARM INCOME OR (LOSS) (ATTACH SCHEDULE F)                   19          .00
20  OTHER                                                      20          .00
21  TOTAL INCOME                                               21     20400.00

ADJUSTMENTS TO INCOME
22  MOVING EXPENSE (ATTACH FORM 3903)           22        .00
23  EMPLOYEE BUS EXP (ATTACH FORM 2106)         23        .00
24  RETIREMENT ARR. (ATTACH FORM 5329)          24        .00
25  PAYMENTS TO KEOGH                           25        .00
26  FORFEITED INTEREST PENALTY                  26        .00
27  ALIMONY PAID                                27        .00
28  TOTAL ADJUSTMENTS                                              28          .00
29  SUBTRACT LINE 28 FROM LINE 21                                  29     20400.00
30  SICK PAY (ATTACH FORM 2440)                                    30          .00
31  ADJUSTED GROSS INCOME                                          31     20400.00

TAX COMPUTATION
32  AMOUNT FROM LINE 31                                            32     20400.00
33  ITEMIZED EXCESS (SCHEDULE A LINE 41)                           33      2559.20
34  TAX TABLE INCOME                                               34     17840.80
35  TAX                                                            35      2174.00
36  ADDITIONAL TAXES FROM FORM 4970 FORM 4972 FORM 5544
    FORM 5405 SECTION 72(M)(5) PENALTY TAX                         36          .00
37  TOTAL                                                          37      2174.00

CREDITS
38  CONTRIBUTIONS TO CANDIDATES                 38        .00
39  EDERLY (ATTACH SCHEDULE R&RP)               39        .00
40  CHIL AND DEPENDENT CARE                     40        .00
41  INVESTMENT (ATTACH FORM 3468)               41        .00
42  FOREIGN (ATTACH FORM 1116)                  42        .00
43  WORK INCENTIVE (ATTACH FORM 4874)           43        .00
44  NEW JOB (ATTACH FORM 5884)                  44        .00
45  SEE INSTRUCTIONS                            45        .00
46  TOTAL CREDITS                                                  46          .00
47  BALANCE                                                        47      2174.00

OTHER TAXES
48  SELF EMPLOYMENT (ATTACH SCHEDULE SE)                           48          .00
49  MINIMUM (ATTACH FORM 4625)                                     49          .00
50  TAX FROM PRIOR YEAR INV CREDIT (FORM 4255)                     50          .00
```

Figure 5 - continued

```
51  FICA ON TIPS (FORM 4137)                                        51         .00
52  UNCOLLECTED FICA                                                52         .00
53  TAX ON RETIREMENT (FORM 5329)                                   53         .00
54  TOTAL TAX                                                       54     2174.00
PAYMENTS
55  TOTAL FED TAX WITHHELD                            55    4200.00
56  ESTIMATED TAX PAYMENTS                            56        .00
57  EARNED INCOME CREDIT                              57        .00
58  AMOUNT PAID WITH FORM 4868                        58        .00
59  EXCESS FICIA                                      59        .00
60  CREDIT FOR FEDERAL TAX ON FUEL (FORM 4136)        60        .00
61  CREDIT FOR REGULATED INVEST. (FORM 2439)          61        .00
 3A SPECIAL                                           3A        .00
62  TOTAL PAYMENTS                                                  62     4200.00

REFUND OR DUE
63  OVERPAID                                          63    2026.00
64  REFUND                                            64    2026.00
65  CREDIT TO 1978                                    65        .00
66  BALANCE DUE                                                     66         .00
```

Tax laws allow u
in pennies, the met
computer, or rou
whole dc

Figure 6 - Completed Schedule A

```
MEDICAL & DENTAL EXP                          CONTRIBUTIONS
 1  INS PREM                         150.00   21   CASH WITH EVID               100.00
 2  MEDICINE                            .00   211  OTHER CASH                      .00
 3  1% OF GROSS                      204.00   212  UNITED F.-                    52.00
 4  LINE 2-LINE3                        .00   213  ----------                      .00
 5  BALANCE OF INS.                     .00   214  ----------                      .00
 6  OTHER MEDICAL                       .00   22   OTHER THAN CASH                 .00
 61 DOCTORS,ETC                         .00   23   CARRYOVER                       .00
 62 HOSPITALS                           .00   24   TOTAL CONTRIBUTIONS          152.00
 63 OTHER                               .00   CASUALTY
 7  TOTAL                               .00   25   LOSS BEFORE INS              200.00
 8  3% OF GROSS                      612.00   26   INSURANCE                       .00
 9  LINE7-LINE8                         .00   27   LINE25-LINE26                200.00
10  TOTAL                            150.00   28   $100 MIN LOSS                100.00
TAXES                                         29   TOTAL                        100.00
11  STATE AND LOCAL INC             1600.55   MISCELLANEOUS
12  REAL ESTATE                     2400.55   30   UNION DUES                      .00
13  GASOLINE TAX                     300.55   31   OTHER                           .00
14  SALES TAX                       1000.55   311  ----------                      .00
15  PERSONAL PROPERTY                   .00   32   TOTAL                           .00
16  OTHER                               .00   SUMMARY
161 EXCISE----                        55.00   33   MEDICAL                      150.00
162 ----------                        55.00   34   TAXES                       5357.20
17  TOTAL                           5357.20   35   INTEREST                        .00
INTEREST                                      36   CONTRIBUTIONS                152.00
18  MORTGAGE                            .00   37   CASUALTY                     100.00
19  OTHER                               .00   38   MISCELLANEOUS                   .00
191 ----------                          .00   39   TOTAL DEDUCTIONS            5759.20
192 ----------                          .00   40   STANDARD DEDUCTION          3200.00
193 ----------                          .00   41   EXCESS DEDUCTION            2559.20
20  TOTAL                               .00
```

Figure 7 - Revised Tax

```
TAX RATE FILE ? M-3
DATA FILE ? 1977
 A NUMBER FROM 1 TO 5 AS FILING STATUS ? 2
0 FOR A 24 LINE VIDEO OR 1 FOR THE PRINTER ? 0

ID# ACTION
=== ===================
  1 FORM 1040
  2 SCHEDULE A
  3 TAX
  4 TAX IF PENNIES ARE ROUNDED
  5 PRINT BLANK 1040
  6 BLANK SCHEDULE A
ID#, 1 TO INPUT OR 0 TO REVIEW ? 1,1

ENTER 0,0,0 TO RETURN.  OTHER ENTRIES ARE FORMATTED 'A,B,C'
WHERE A=LINE NUMBER OF ENTRY
      B= 0 TO OVERRIDE PRIOR ENTRY OR 1 TO BE ADDED TO PRIOR
      C= THE AMOUNT
?8,1,10
?0,0,0

ID# ACTION
=== ===================
  1 FORM 1040
  2 SCHEDULE A
  3 TAX
  4 TAX IF PENNIES ARE ROUNDED
  5 PRINT BLANK 1040
  6 BLANK SCHEDULE A
ID#, 1 TO INPUT OR 0 TO REVIEW ? 3,0

YOUR TAX IS  $2186.00
YOU PAID     $4200.00
REFUND IS    $2014.00
```

detects the first entry, an illegal line number (all line numbers are verified). The rest of the data follows. (Note: the computer program requests a description of the deduction where applicable.)

$ 150.55	insurance premiums for medical care
1,600.55	state and local income tax withholdings
2,400.55	real estate tax
300.55	gasoline tax
1,000.55	sales tax
55.00	excise tax
100.00	cash contribution with evidence
52.00	cash contribution
200.00	casualty loss

Armed with the data just supplied, the computer dashes off and calculates your tax return. Figures 5 and 6 show completed Form 1040 and Schedule A. Notice how the computer followed all tax regulations. For example, the dividend exclusion was subtracted from total dividends, the medical insurance premium was reduced to the maximum $150, the casualty loss was handled properly and your tax was calculated exactly according to the new tax tables.

Now let's assume you made a mistake and entered wages as $20,000 instead of the $20,010 indicated on your W-2 Form. Figure 7 shows how to add this correction to the data already on file. Enter the $10 difference and then look at our new tax by selecting Action 3. The additional $10 raises the tax bill by $12—we have been foiled by the new tax tables; our prior taxable income of $17,840.80 was only $9.20 below the next highest tax bracket. Can the computer do anything legal to help us? It can try. Tax laws allow us to leave all figures in pennies (the method

o leave all figures
)d employed by the
l them to nearest
ar values.

employed by the computer) or round them to nearest whole dollar values. By entering Action 4 the computer then re-calculates the entire return, rounding to nearest whole dollars for each item. This procedure works and your $5,000 computer will save you $12, or just enough to cover the electricity consumed.

Figure 8 shows the revised section of Form 1040. This was executed specifing a video terminal and resulting in the computer asking "ready?" Before the next screen prints at 19,200 baud, we have time to study the results. (A return or any input causes the next screen to be presented.)

The Tax Rate file used with the disk system contains 696 values and is 14 blocks long. Figure 9 gives the program used to generate this information. If you don't have a disk, this data can be entered in an added subroutine, although the additional memory modification will cost you money. (A sample run of this program is also given.)

Line 95 exists for demonstration purposes, so you wouldn't have to see all 696 values printed. This run is for a married person with 4 deductions. Federal Tax Rate Schedule and Table values shown in Figure 10 correspond to the values generated in Figure 9. Line 1,000 of the program represents the values marked on the Federal Tax Rate Schedule in Figure 10. Line 1010 represents the tax on that amount and line 1020 gives the rate to be charged on excess earnings. These lines must be edited to calculate single returns and on returns filed separately by married couples.

Figure 11 details the creation of a blank data file. The data is very basic (pardon the pun):

Figure 7 - continued

```
ID# ACTION
=== ===================
  1 FORM 1040
  2 SCHEDULE A
  3 TAX
  4 TAX IF PENNIES ARE ROUNDED
  5 PRINT BLANK 1040
  6 BLANK SCHEDULE A
ID#, 1 TO INPUT OR 0 TO REVIEW ? 4,0

YOUR TAX IS  $2174.00
YOU PAID     $4200.00
REFUND IS    $2026.00
```

Figure 8 - Updated Form 1040

```
ID# ACTION
=== ===================
  1 FORM 1040
  2 SCHEDULE A
  3 TAX
  4 TAX IF PENNIES ARE ROUNDED
  5 PRINT BLANK 1040
  6 BLANK SCHEDULE A
ID#, 1 TO INPUT OR 0 TO REVIEW ? 1,0

1040 U.S. INDIVIDUAL INCOME TAX RETURN
INCOME
 8  WAGES FROM W2 FORM                                              8     20010.00
 9  INTEREST INCOME (IF OVER $400 ATTACH SCHED B)                   9          .00
 1A DIVINDENDS      300.00  2A LESS EXCL.    200.00 BALANCE        10C       100.00
11  STATE AND LOCAL TAX REFUNDS                                    11        300.00
12  ALIMONY RECEIVED                                               12           .00
13  BUSINESS INCOME/(LOSS) (ATTACH SCHEDULE C)                     13           .00
14  CAPITAL GAIN/(LOSS) (ATTACH SCHEDULE D)                        14           .00
15  50% OF CAPITAL GAIN DISTRIBUTIONS                              15           .00
16  NET GAIN OR (LOSS) FROM SUPPL. SCHED.(ATTACH FORM 4797)        16           .00
17  FULLY TAXABLE PENSIONS AND ANNUNITIES NOT ON SCHED E           17           .00
18  PENSIONS, ANNUNITIES, RENT, ETC. (ATTACH SCHED E)              18           .00
19 FARM INCOME OR (LOSS) (ATTACH SCHEDULE F)                       19           .00
20  OTHER                                                          20           .00
21  TOTAL INCOME                                                   21      20410.00

RETURN WHEN READY FOR REST OF SCHED ?
```

Figure 9 - Tax Rate Program

```
5 DIMR(3,13)
10 FORA=1TO3\FORB=1TO13\READR(A,B)\IFA=1THENR(A,B)=R(A,B)*100
20 IFA=3THENR(A,B)=R(A,B)*.01\NEXT\NEXT
25 INPUT'FILE NAME ? ',Z$
30 OPEN#0,Z$
40 INPUT'NUMBER OF DEDUCTIONS ? ',Z9
45 F=35*Z9
50 Z9=Z9*750
60 FORA=1TO5\!'   BASE    TAX',\NEXT\!''
65 FORA=1TO70\!'=',\NEXT\!' '
90 FORA=1TO696\B=(A*50)+5175-Z9
100 FORC=1TO13\IFB< R(1,C)THENEXIT110\NEXT\D=0\GOTO115
110 C=C-1
111 D=R(2,C)+((B-R(1,C))*R(3,C))
112        G=(B-3200)*.02\IFG>180THENG=180\IFG>FTHENF=G
114 D=D-F\IFD<0THEND=0
115 E=5150+(A*50)
116 D=INT(D+.5)
117 !%7I,B+Z9-25,D,\V=V+1\IFV<5THEN120\V=0\!''
120 WRITE#0,D\NEXT\CLOSE#0
1000 DATA32,42,52,62,72,112,152,192,232,272,312,352,392
1010 DATA0,140,290,450,620,1380,2260,3260,4380,5660,7100,8660,10340
1020 DATA14,15,16,17,19,22,25,28,32,36,39,42,45
READY
95 IFA>64THENEND
RUN

FILE NAME ? M-4
NUMBER OF DEDUCTIONS ? 4
   BASE     TAX   BASE     TAX   BASE     TAX   BASE     TAX   BASE     TAX
======================================================================
   5200       0   5250       0   5300       0   5350       0   5400       0
   5450       0   5500       0   5550       0   5600       0   5650       0
   5700       0   5750       0   5800       0   5850       0   5900       0
   5950       0   6000       0   6050       0   6100       0   6150       0
   6200       0   6250       0   6300       0   6350       0   6400       0
   6450       0   6500       0   6550       0   6600       0   6650       0
   6700       0   6750       0   6800       0   6850       0   6900       0
   6950       0   7000       0   7050       0   7100       0   7150       0
   7200       4   7250      11   7300      19   7350      26   7400      34
   7450      41   7500      49   7550      56   7600      64   7650      71
   7700      79   7750      86   7800      94   7850     101   7900     109
   7950     116   8000     124   8050     131   8100     139   8150     146
   8200     154   8250     162   8300     170   8350     178
READY
```

Figure 10 - Government Tax Tables

If line 34, Form 1040 is—		And the total number of exemptions claimed on line 7 is—							
Over	But not over	2	3	4	5	6	7	8	9
		Your tax is—							
If $5,200 or less your tax is 0									
5,200	5,250	4	0	0	0	0	0	0	0
5,250	5,300	11	0	0	0	0	0	0	0
5,300	5,350	18	0	0	0	0	0	0	0
5,350	5,400	25	0	0	0	0	0	0	0
5,400	5,450	32	0	0	0	0	0	0	0
5,450	5,500	39	0	0	0	0	0	0	0
5,500	5,550	46	0	0	0	0	0	0	0
5,550	5,600	53	0	0	0	0	0	0	0
5,600	5,650	60	0	0	0	0	0	0	0
5,650	5,700	67	0	0	0	0	0	0	0
5,700	5,750	74	0	0	0	0	0	0	0
5,750	5,800	81	0	0	0	0	0	0	0
5,800	5,850	89	0	0	0	0	0	0	0
5,850	5,900	96	0	0	0	0	0	0	0
5,900	5,950	104	0	0	0	0	0	0	0
5,950	6,000	111	0	0	0	0	0	0	0
6,000	6,050	119	0	0	0	0	0	0	0
6,050	6,100	126	0	0	0	0	0	0	0
6,100	6,150	134	0	0	0	0	0	0	0
6,150	6,200	141	0	0	0	0	0	0	0
6,200	6,250	149	4	0	0	0	0	0	0
6,250	6,300	156	11	0	0	0	0	0	0
6,300	6,350	164	18	0	0	0	0	0	0
6,350	6,400	171	25	0	0	0	0	0	0
6,400	6,450	179	32	0	0	0	0	0	0
6,450	6,500	186	39	0	0	0	0	0	0
6,500	6,550	194	46	0	0	0	0	0	0
6,550	6,600	201	54	0	0	0	0	0	0
6,600	6,650	209	61	0	0	0	0	0	0
6,650	6,700	216	69	0	0	0	0	0	0
6,700	6,750	224	76	0	0	0	0	0	0
6,750	6,800	232	84	0	0	0	0	0	0
6,800	6,850	240	91	0	0	0	0	0	0
6,850	6,900	248	99	0	0	0	0	0	0
6,900	6,950	256	106	0	0	0	0	0	0
6,950	7,000	264	114	0	0	0	0	0	0
7,000	7,050	272	121	0	0	0	0	0	0
7,050	7,100	280	129	0	0	0	0	0	0
7,100	7,150	288	136	0	0	0	0	0	0
7,150	7,200	296	144	0	0	0	0	0	0
7,200	7,250	304	151	4	0	0	0	0	0
7,250	7,300	312	159	11	0	0	0	0	0
7,300	7,350	320	166	19	0	0	0	0	0
7,350	7,400	328	174	26	0	0	0	0	0
7,400	7,450	336	181	34	0	0	0	0	0
7,450	7,500	344	189	41	0	0	0	0	0
7,500	7,550	352	197	49	0	0	0	0	0
7,550	7,600	360	205	56	0	0	0	0	0
7,600	7,650	368	213	64	0	0	0	0	0
7,650	7,700	376	221	71	0	0	0	0	0
7,700	7,750	384	229	79	0	0	0	0	0
7,750	7,800	393	237	86	0	0	0	0	0
7,800	7,850	401	245	94	0	0	0	0	0
7,850	7,900	410	253	101	0	0	0	0	0
7,900	7,950	418	261	109	0	0	0	0	0
7,950	8,000	427	269	116	0	0	0	0	0
8,000	8,050	435	277	124	0	0	0	0	0
8,050	8,100	444	285	131	0	0	0	0	0
8,100	8,150	452	293	139	0	0	0	0	0
8,150	8,200	461	301	146	0	0	0	0	0
8,200	8,250	469	309	154	6	0	0	0	0
8,250	8,300	476	317	162	14	0	0	0	0
8,300	8,350	484	325	170	21	0	0	0	0
8,350	8,400	491	333	178	29	0	0	0	0

66 variables for Form 1040
54 variables for Schedule A
90 string characters for Schedule A

This file is only 3 blocks long. You can avoid the file by storing data in free memory using PEEK and POKE (some BASICS call it FILL and EXAM) or by tape storage procedures.

If all else fails, these programs can be modified for other systems by either eliminating the tax table read (a random access read is currently used) and looking up and inputing the tax yourself; or eliminating input proce-

Married Filing Joint Returns and Qualifying Widows and Widowers

Use this schedule if you checked **Box 2** or **Box 5** on Form 1040—

If the amount on Schedule TC, Part I, line 3, is:		Enter on Schedule TC, Part I, line 4:	
Not over $3,200............		—0—	
Over—	But not over—		of the amount over—
$3,200	$4,200	14%	$3,200
$4,200	$5,200	$140+15%	$4,200
$5,200	$6,200	$290+16%	$5,200
$6,200	$7,200	$450+17%	$6,200
$7,200	$11,200	$620+19%	$7,200
$11,200	$15,200	$1,380+22%	$11,200
$15,200	$19,200	$2,260+25%	$15,200
$19,200	$23,200	$3,260+28%	$19,200
$23,200	$27,200	$4,380+32%	$23,200
$27,200	$31,200	$5,660+36%	$27,200
$31,200	$35,200	$7,100+39%	$31,200
$35,200	$39,200	$8,660+42%	$35,200
$39,200	$43,200	$10,340+45%	$39,200
$43,200	$47,200	$12,140+48%	$43,200
$47,200	$55,200	$14,060+50%	$47,200
$55,200	$67,200	$18,060+53%	$55,200
$67,200	$79,200	$24,420+55%	$67,200
$79,200	$91,200	$31,020+58%	$79,200
$91,200	$103,200	$37,980+60%	$91,200
$103,200	$123,200	$45,180+62%	$103,200
$123,200	$143,200	$57,580+64%	$123,200
$143,200	$163,200	$70,380+66%	$143,200
$163,200	$183,200	$83,580+68%	$163,200
$183,200	$203,200	$97,180+69%	$183,200
$203,200	....	$110,980+70%	$203,200

dures by letting input take the form of data lines in the actual program. This way, no matter how you store the program you retain the data.

The variables in the Form 1040 section are labeled "v" and are dimensioned to the 66th. Numbers with ".00" after them, in Figure 2, show you what each represents. In Schedule A, use variable "d" to the 54th and note the details in Figure 3. Figure 12 describes the program functions by line numbers.

Good luck.

Figure 11 - Data File Program

```
5 DIMN$(90)
10 INPUT"FILE ? ",Z$\OPEN#0,Z$
20 FORA=1TO9\N$="----------"+N$\NEXT
30 FORA=1TO120\WRITE#0,B\
   NEXT\WRITE#0,N$
40 CLOSE#0\END
READY
RUN

FILE ? 1977
READY
```

Figure 12 - Program Functions by Line Number

1-45	Dimension variables and solicit initial data
50-90	Subroutine to read data file
92-96	Ask remaining opening questions.
100-320	Print action codes, ask for action and branch to appropriate section of the program
1000-1670	Print Form 1040
1700-1760	Input for Form 1040
1800-2120	Print Schedule A
2130-2230	Input for Schedule A
2300-2300	Print Tax only
2350-2370	Round all figures to whole dollars
2400-2450	Subroutine for printing taxes only
2500-2600	Set all values to variable dimensions
7000-7010	Ask for "return" on video terminal prints
7100-7230	Forms 1040 calculations
7300-7398	Schedule A calculations
9000-9999	Save date file and end program

Main Program

```
1 DIMV(66),D(54),D$(90)
6 E(1)=2200\E(2)=3200\E(3)=1600\E(4)=2200\E(5)=3200
10 LINE79
20 INPUT"TAX RATE FILE ? ",Z$\OPEN#1,Z$
40 INPUT"DATA FILE ? ",X$
45 GOSUB50\GOTO92
50 OPEN#0,X$
55 FORA=1TO66\READ#0,V(A)\NEXT
60 FORA=1TO54\READ#0,D(A)\NEXT\READ#0,D$(1,90)
90 CLOSE#0\RETURN
92 INPUT" A NUMBER FROM 1 TO 5 AS FILING STATUS ? ",T
94 IFT<1ORT>5THEN92
96  INPUT"0 FOR A 24 LINE VIDEO OR 1 FOR THE PRINTER ? ",S3\!""\!""
100 !""\!""
105 !"ID# ACTION"\!"=== ===================="
110 !"  1 FORM 1040"\!"  2 SCHEDULE A"
120 !"  3 TAX"\!"  4 TAX IF PENNIES ARE ROUNDED"
130 !"  5 PRINT BLANK 1040"
140 !"  6 BLANK SCHEDULE A"
300 INPUT"ID#, 1 TO INPUT OR 0 TO REVIEW ? ",S1,S4
310 IFS1>6THEN100\IFS1<1THEN9000
320 IFS1> 9THEN330\ONS1GOTO1000,1800,2300,2350,2500,2600,9000,9000,9000
1000 IFS4=1THEN1700\IFS3=1THENS2=9
1005 !               ""\!"1040 U.S. INDIVIDUAL INCOME TAX RETURN"\!"INCOME"
1010 !" 8  WAGES FROM W2 FORM",TAB(65)," 8  ",%10F2,V(8)
1020 !" 9  INTEREST INCOME (IF OVER $400 ATTACH SCHED B)",TAB(65)," 9  ",
1025 !%10F2,V(9)
1030 !" 1A DIVINDENDS",%10F2,V(1),"  2A LESS EXCL.",V(2)," BALANCE",TAB(65),
1035 !"10C ",%10F2,V(10)
1040 !"11  STATE AND LOCAL TAX REFUNDS",TAB(65),"11  ",%10F2,V(11)
1050 !"12  ALIMONY RECEIVED",TAB(65),"12  ",%10F2,V(12)
1060 !"13  BUSINESS INCOME/(LOSS) (ATTACH SCHEDULE C)",TAB(65),"13  ",
1065 !%10F2,V(13)
1070!"14  CAPITAL GAIN/(LOSS) (ATTACH SCHEDULE D)",TAB(65),"14  ",
1075 !%10F2,V(14)
1080 !"15  50% OF CAPITAL GAIN DISTRIBUTIONS",TAB(65),"15  ",%10F2,V(15)
1090 !"16  NET GAIN OR (LOSS) FROM SUPPL. SCHED.(ATTACH FORM 4797)",
1095 !TAB(65),"16  ",%10F2,V(16)
1100 !"17  FULLY TAXABLE PENSIONS AND ANNUNITIES NOT ON SCHED E",
1105 !TAB(65),"17  ",%10F2,V(17)
1110 !"18  PENSIONS, ANNUNITIES, RENT, ETC. (ATTACH SCHED E)",
1115 !TAB(65),"18  ",%10F2,V(18)
1120 !"19 FARM INCOME OR (LOSS) (ATTACH SCHEDULE F)",TAB(65),"19  ",
1125 !%10F2,V(19)
1130 !"20  OTHER",TAB(65),"20  ",%10F2,V(20)
1140 !"21  TOTAL INCOME",TAB(65),"21  ",%10F2,V(21)
1150 GOSUB7000
1160 !" "\!"ADJUSTMENTS TO INCOME"
1170 !"22  MOVING EXPENSE (ATTACH FORM 3903)",TAB(50),"22  ",%10F2,V(22)
1180 !"23  EMPLOYEE BUS EXP (ATTACH FORM 2106)",TAB(50),"23  ",%10F2,V(23)
1190 !"24  RETIREMENT ARR. (ATTACH FORM 5329)",TAB(50),"24  ",%10F2,V(24)
1200 !"25  PAYMENTS TO KEOGH",TAB(50),"25  ",%10F2,V(25)
1210 !"26  FORFEITED INTEREST PENALTY",TAB(50),"26  ",%10F2,V(26)
1220 !"27  ALIMONY PAID",TAB(50),"27  ",%10F2,V(27)
1230 !"28  TOTAL ADJUSTMENTS",TAB(65),"28  ",%10F2,V(28)
1240 !"29  SUBTRACT LINE 28 FROM LINE 21",TAB(65),"29  ",%10F2,V(29)
1250 !"30  SICK PAY (ATTACH FORM 2440)",TAB(65),"30  ",%10F2,V(30)
1260 !"31  ADJUSTED GROSS INCOME",TAB(65),"31  ",%10F2,V(31)
1270 GOSUB7000
1280 !""\!"TAX COMPUTATION"
1285 !"32  AMOUNT FROM LINE 31",TAB(65),"32  ",%10F2,V(32)
1290 !"33  ITEMIZED EXCESS (SCHEDULE A LINE 41)",TAB(65),"33  ",%10F2,V(33)
1295 !"34  TAX TABLE INCOME",TAB(65),"34  ",%10F2,V(34)
1300 !"35  TAX",TAB(65),"35  ",%10F2,V(35)
1310 !"36  ADDITIONAL TAXES FROM FORM 4970 FORM 4972 FORM 5544"
1320 !"    FORM 5405 SECTION 72(M)(5) PENALTY TAX",TAB(65),"36  ",
1325 !%10F2,V(36)
1327 !"37  TOTAL",TAB(65),"37  ",%10F2,V(37)
1330 !""\!"CREDITS"
1340 !"38  CONTRIBUTIONS TO CANDIDATES",TAB(50),"38  ",%10F2,V(38)
1350 !"39  EDERLY (ATTACH SCHEDULE R&RP)",TAB(50),"39  ",%10F2,V(39)
1360 !"40  CHIL AND DEPENDENT CARE",TAB(50),"40  ",%10F2,V(40)
1370 !"41  INVESTMENT (ATTACH FORM 3468)",TAB(50),"41  ",%10F2,V(41)
1380 !"42  FOREIGN (ATTACH FORM 1116)",TAB(50),"42  ",%10F2,V(42)
1390 !"43  WORK INCENTIVE (ATTACH FORM 4874)",TAB(50),"43  ",%10F2,V(43)
1400 !"44  NEW JOB (ATTACH FORM 5884)",TAB(50),"44  ",%10F2,V(44)
1410 !"45  SEE INSTRUCTIONS",TAB(50),"45  ",%10F2,V(45)
1420 !"46  TOTAL CREDITS",TAB(65),"46  ",%10F2,V(46)
1430 !"47  BALANCE",TAB(65),"47  ",%10F2,V(47)
1435 GOSUB7000
1440 !""\!"OTHER TAXES"
1450 !"48  SELF EMPLOYMENT (ATTACH SCHEDULE SE)",TAB(65),"48  ",
1455 !%10F2,V(48)
```

(Continued on following page)

If all else fails, these programs can be modified for other systems by either eliminating the tax table read and looking up and inputting the tax yourself; or eliminating input procedures by letting input take the form of data lines in the active program.

```
1460 !"49  MINIMUM (ATTACH FORM 4625)",TAB(65),"49  ",%10F2,V(49)
1470!"50  TAX FROM PRIOR YEAR INV CREDIT (FORM 4255)",
1475 !TAB(65),"50  ",%10F2,V(50)
1480 !"51  FICA ON TIPS (FORM 4137)",TAB(65),"51  ",%10F2,V(51)
1490 !"52  UNCOLLECTED FICA",TAB(65),"52  ",%10F2,V(52)
1500 !"53  TAX ON RETIREMENT (FORM 5329)",TAB(65),"53  ",%10F2,V(53)
1510 !"54  TOTAL TAX",TAB(65),"54  ",%10F2,V(54)
1515 IFS3<>1THEN1520\FORA=1TO8\!"   "\NEXT
1520 GOSUB7000\!""\!"PAYMENTS"
1530 !"55  TOTAL FED TAX WITHHELD",TAB(50),"55  ",%10F2,V(55)
1540 !"56  ESTIMATED TAX PAYMENTS",TAB(50),"56  ",%10F2,V(56)
1550 !"57  EARNED INCOME CREDIT",TAB(50),"57  ",%10F2,V(57)
1560 !"58  AMOUNT PAID WITH FORM 4868" ,TAB(50),"58  ";%10F2,V(58)
1570 !"59  EXCESS FICIA",TAB(50),"59  ",%10F2,V(59)
1580 !"60  CREDIT FOR FEDERAL TAX ON FUEL (FORM 4136)",TAB(50),"60  ",
1585 !%10F2,V(60)
1590 !"61  CREDIT FOR REGULATED INVEST. (FORM 2439)",TAB(50),"61  ",
1595 !%10F2,V(61)
1600 !" 3A SPECIAL",TAB(50)," 3A ",%10F2,V(3)
1610 !"62  TOTAL PAYMENTS",TAB(65),"62  ",%10F2,V(62)
1620 !""\!"REFUND OR DUE"
1630 !"63  OVERPAID",TAB(50),"63  ",%10F2,V(63)
1640 !"64  REFUND",TAB(50),"64  ",%10F2,V(63)
1650 !"65  CREDIT TO 1978",TAB(50),"65  ",%10F2,V(65)
1660 !"66  BALANCE DUE",TAB(65),"66  ",%10F2,V(66)
1670 GOSUB7000\IFS1=5THENGOSUB50\GOTO100
1700 !""\!"ENTER 0,0,0 TO RETURN.  OTHER ENTRIES ARE FORMATTED 'A,B,C'"
1710 !"WHERE A=LINE NUMBER OF ENTRY"
1720 !"       B= 0 TO OVERRIDE PRIOR ENTRY OR 1 TO BE ADDED TO PRIOR"
1730 !"       C= THE AMOUNT"
1740 INPUTB,A,C\IFB<>0THEN1750\GOSUB7100\GOSUB7300\GOTO100
1745 !"ILLEGAL ENTRY"\GOTO1740
1750 IFB>3ANDB<8THEN1745\IFB<1ORB>66THEN1745
1755 C=INT(C*100)\C=C*.01
1760 IFA=0THENV(B)=C\IFA<>0THENV(B)=V(B)+C\GOTO1740
1800 IFS4=1THEN2130\!""\!"SCHEDULE A"
1810 !"MEDICAL & DENTAL EXP",TAB(40),"CONTRIBUTIONS"
1820 !" 1  INS PREM",TAB(28),%10F2,D(1),"  21  CASH WITH EVID",
1825 !TAB(68),%10F2,D(21)
1830 !" 2  MEDICINE",TAB(28),%10F2,D(2),"  211 OTHER CASH",
1835 !TAB(68),%10F2,D(50)
1840!" 3  1% OF GROSS",TAB(28),%10F2,D(3),"  212 ",D$(51,60),
1845 !TAB(68),%10F2,D(51)
1850!" 4  LINE 2-LINE3",TAB(28),%10F2,D(4),"  213 ",D$(61,70),
1855 !TAB(68),%10F2,D(52)
1860 !" 5  BALANCE OF INS.",TAB(28),%10F2,D(5),"  214 ",D$(71,80),
1865 !TAB(68),%10F2,D(53)
1870 !" 6  OTHER MEDICAL",TAB(28),%10F2,D(6),"  22  OTHER THAN CASH",
1875 !TAB(68),%10F2,D(22)
1880 !" 61 DOCTORS,ETC",TAB(28),%10F2,D(42),"  23  CARRYOVER",
1885 !TAB(68),%10F2,D(23)
1890 !" 62 HOSPITALS",TAB(28),%10F2,D(43),"  24  TOTAL CONTRIBUTIONS",
1895 !TAB(68),%10F2,D(24)
1900 !" 63 OTHER",TAB(28),%10F2,D(44),"  CASUALTY"
1910 !" 7  TOTAL",TAB(28),%10F2,D(7),"  25  LOSS BEFORE INS",
1915 !TAB(68),%10F2,D(25)
1920 !" 8  3% OF GROSS",TAB(28),%10F2,D(8),"  26  INSURANCE",
1925 !TAB(68),%10F2,D(26)
1930 !" 9  LINE7-LINE8",TAB(28),%10F2,D(9),"  27  LINE25-LINE26",
1935 !TAB(68),%10F2,D(27)
1940 !"10  TOTAL",TAB(28),%10F2 ,D(10),"  28  $100 MIN LOSS",
1945 !TAB(68),%10F2,D(28)
1950 !"TAXES",TAB(40),"29  TOTAL",TAB(68),%10F2,D(29)
1960 !"11  STATE AND LOCAL INC",TAB(28),%10F2,D(11),"  MISCELLANEOUS"
1970 !"12  REAL ESTATE",TAB(28),%10F2,D(12),"  30  UNION DUES",
1975 !TAB(68),%10F2,D(30)
1980 !"13  GASOLINE TAX",TAB(28),%10F2,D(13),"  31  OTHER",
1985 !TAB(68),%10F2,D(31)
```

```
1990 !"14  SALES TAX",TAB(28),%10F2,D(14),"  311 ",D$(81,90),
1995 !TAB(68),%10F2,D(54)
2000 !"15  PERSONAL PROPERTY",TAB(28),%10F2,D(15),"  32  TOTAL",
2005 !TAB(68),%10F2,D(32)
2010 GOSUB7000
2020 !"16  OTHER",TAB(28),%10F2,D(16),"  SUMMARY"
2030 !"161 ",D$(1,10),TAB(28),%10F2,D(45),"  33  MEDICAL",
2035 !TAB(68),%10F2,D(33)
2040 !"162 ",D$(11,20),TAB(28),%10F2,D(45),"  34  TAXES",
2045 !TAB(68),%10F2,D(34)
2050 !"17  TOTAL",TAB(28),%10F2,D(17),"  35  INTEREST",
2055 !TAB(68),%10F2,D(35)
2060 !"INTEREST",TAB(40),"36  CONTRIBUTIONS",
2065 !TAB(68),%10F2,D(36)
2070 !"18  MORTGAGE",TAB(28),%10F2,D(18),"  37  CASUALTY",
2075 !TAB(68),%10F2,D(37)
2080 !"19  OTHER",TAB(28),%10F2,D(19),"  38  MISCELLANEOUS",
2085 !TAB(68),%10F2,D(38)
2090 !"191 ",D$(21,30),TAB(28),%10F2,D(47),"  39  TOTAL DEDUCTIONS",
2095 !TAB(68),%10F2,D(39)
2100 !"192 ",D$(31,40),TAB(28),%10F2,D(48),"  40  STANDARD DEDUCTION",
2105 !TAB(68),%10F2,D(40)
2110 !"193 ",D$(41,50),TAB(28),%10F2,D(49),"  41  EXCESS DEDUCTION",
2115 !TAB(68),%10F2,D(41)
2120 !"20  TOTAL",TAB(28),%10F2,D(20)\IFS1=6THENGOSUB50\GOTO100
2130 !""\!"ENTER 0,0,0 TO RETURN.  OTHER ENTRIES ARE FORMATTED 'A,B,C'"
2140 !"WHERE A=LINE NUMBER OF ENTRY"
2150 !"       B= 0 TO OVERRIDE PRIOR ENTRY OR 1 TO BE ADDED TO PRIOR"
2160 !"       C= THE AMOUNT"
2170 INPUTA,B,C\IFA<>0THEN2180\GOSUB7300\GOSUB7100\GOTO100
2175 !"ILLEGAL ENTRY"\GOTO2170
2180 IFA<0THEN2175\IFA<42THEN2200
2185 IFA>60ANDA<64THENA=A-19
2190 IFA>160ANDA<163THENA=A-116
2195 IFA>210ANDA<215THENA=A-161
2197 IFA>190ANDA<194THENA=A-144
2199 IFA=311THENA=54
2200 IFA>54THEN2175
2205 C=INT(C*100)\C=C*.01
2210 IFB=0THEND(A)=C\IFB<>0THEND(A)=D(A)+C
2220 IFA<45THEN2170\Z$=".........."\INPUT"DESCRIPTION ? ",Z$
2225 A=A-44\IFA=6THEN2170\IFA>5THENA=A-1
2230 A=A*10-9\D$(A,A+9)=Z$\GOTO2170
2300 GOSUB2400\GOTO100
2350 FORA=1TO54\D(A)=INT(D(A)+.5)\NEXT
2360 FORA=1TO66\V(A)=INT(V(A)+.5)\NEXT
2370 GOSUB7300\GOSUB7100\GOSUB7300\GOSUB2400\GOTO100
2400 !""
2410 !"YOUR TAX IS",%$10F2,V(54)
2420 !"YOU PAID   ",%$10F2,V(62)
2430 IFV(64)=0THEN2450
2440 !"REFUND IS  ",%$10F2,V(64)\!""\RETURN
2450 !"YOU OWE    ",%$10F2,V(66)\!""\RETURN
2500 FORA=1TO66\V(A)=A\NEXT\GOTO1005
2600 FORA=1TO54\D(A)=A\NEXT\S4=2\GOTO1800
7000 IFS3=1THENRETURN\!""\INPUT"RETURN WHEN READY FOR REST OF SCHED ?",Z$
7010 RETURN
7100 V(10)=V(1)-V(2)\IFV(10)<0THENV(10)=0
7110 V(21)=0\FORA=8TO20\V(21)=V(21)+V(A)\NEXT
7120 V(28)=0\FORA=22TO27\V(28)=V(28)+V(A)\NEXT
7130 V(29)=V(21)-V(28)\V(31)=V(29)-V(30)\V(32)=V(31)
7140 V(34)=V(32)-V(33)\V(35)=INT((V(34)-5150)/50)
7150 IFV(35)>1THEN7160\V(35)=0\GOTO7180
7160 IFV(35)<697THEN7170\V(35)=99999.99\GOTO7180
7170 A=V(35)*5-5\READ#1%A,V(35)
7180 V(37)=V(35)+V(36)\V(46)=0\FORA=38TO45\V(46)=V(46)+V(A)\NEXT
7190 V(47)=V(37)-V(46)\V(54)=0\FORA=47TO53\V(54)=V(54)+V(A)\NEXT
7200 V(62)=V(3)\FORA=55TO61\V(62)=V(62)+V(A)\NEXT
7210 V(63)=0\IFV(54)>V(62)THEN7220\V(63)=V(62)-V(54)
7220 V(64)=V(63)\V(65)=0\V(66)=0\IFV(63)=0THENV(66)=V(54)-V(62)
7230 RETURN
7300 IFD(1)>150THEND(1)=150\D(3)=V(31)*.01\D(4)=D(2)-D(3)
7310 IFD(4)<0THEND(4)=0\D(7)=D(4)+D(5)+D(6)+D(42)+D(43)+D(44)
7320 D(8)=V(31)*.03\D(9)=D(7)-D(8)\IFD(9)<0THEND(9)=0
7330 D(10)=D(1)+D(9)\D(17)=0\FORA=11TO16\D(17)=D(17)+D(A)\NEXT
7340 D(17)=D(17)+D(45)+D(46)\D(20)=D(18)+D(19)+D(47)+D(48)+D(49)
7350 D(24)=D(21)+D(50)+D(51)+D(52)+D(53)+D(22)+D(23)
7360 D(27)=D(25)-D(26)\IFD(27)<0THEND(27)=0
7365 D(28)=D(27)\IF100<D(28)THEND(28)=100\D(29)=D(27)-D(28)
7370 D(32)=D(30)+D(31)+D(54)\D(33)=D(10)\D(34)=D(17)\D(35)=D(20)
7380 D(36)=D(24)\D(37)=D(29)\D(38)=D(32)\D(39)=0
7390 FORA=33TO38\D(39)=D(39)+D(A)\NEXT\D(40)=E(T)
7394 D(41)=D(39)-D(40)\IFD(41)<0THEND(41)=0
7398 V(33)=D(41)\RETURN
9000 INPUT"FILE TO SAVE DATA ? ",Z$\IFZ$=""THEN9999
9005 OPEN#0,Z$
9010 FORA=1TO66\WRITE#0,V(A)\NEXT
9020 FORA=1TO54\WRITE#0,D(A)\NEXT\WRITE#0,D$(1,90)
9990 CLOSE#0
9999 CLOSE#1\!FREE(0)\END
READY
```

Let's Get Personal in Anaheim

June 6-8, 1978

A rewarding personal experience is in store for you June 6-8 at the NCC '78 Personal Computing Festival... the most comprehensive personal computing event ever held. The Festival, a separate feature of the National Computer Conference, will include approximately 30 program sessions, commercial exhibits of consumer computing products and services, plus a contest and exhibit of microprocessor systems and applications. All Festival activities will take place in the Disneyland Hotel Complex, just a few minutes from the Anaheim Convention Center, site of this year's NCC.

Plan now to attend the big, new NCC '78 Personal Computing Festival. The program will include special paper, panel, and tutorial sessions on such topics as speech synthesis and recognition, computerized music systems, hardware and software design, computer graphics, and small business systems. All papers will be published in a softbound volume, *Festival Digest '78,* which will be available during NCC.

Festival exhibits will provide an extensive display of commercial offerings by organizations serving the personal computing field. More than 100 companies, occupying over 175 booths, will display systems, components, terminals, software, kits, disc and tape cassettes, relevant publications, and related hobby items.

Rounding out the Festival will be a contest featuring microprocessor systems, devices, and applications ranging from home-brew DOS and graphics terminals to educational applications and computer games. Prizes will be awarded for the best exhibits.

Don't miss the year's most exciting personal computing event. For more information, return the coupon or call AFIPS at 201/391-9810.

☐ Please keep me up-to-date on Festival plans and activities.

☐ My company is interested in exhibiting at the Festival.

☐ Please send me information on the special NCC Travel Service.

Name ______________________

Company ______________ Division ________

Street ______________________

City ____________ State __________ Zip ______

NCC '78
Personal Computing
Festival

PC

c/o AFIPS, 210 Summit Avenue
Montvale, N.J. 07645
telephone: 201/391-9810

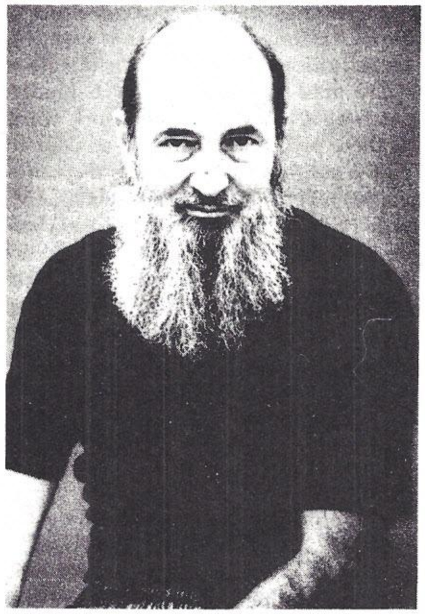
It's Student and Computer, One on One

BY SALLY VALENTE KIESTER

If computer theory is based on physics, why shouldn't physics teaching be based on computers? At the University of California at Irvine, Alfred Bork, professor of physics, has answered this query emphatically: it should. Each quarter, 300 Irvine students study introductory physics at a computer terminal, watching questions form on the screen and tapping out answers. The only resemblance to a conventional course is a weekly summary lecture, a recommended textbook, and a written final exam. While students have frequent contact with Bork and unlimited access to teaching assistants, it's student and machine, one on one.

Bork, a slight, bald figure whose chest-length gray beard makes him look like an unobtrusive Old Testament prophet, considers the marriage of computers and physics instruction an ideal union. "Students learn best by interaction," he says. "Computer-based instruction affords that opportunity on an individual basis. A lecture to 300 persons is totally depersonalized. A textbook must seek a common denominator for all readers. But the computer is so flexible that each student has a unique educational experience. Each one can follow his own pace and concentrate where he needs to."

Physics 3, one of the courses developed under Irvine's Physics Computer Development Project (PCDP), is a three-quarter course primarily for premed juniors and seniors who need the credit from one physics course for medical school admission. The first quarter is totally devoted to computer instruction; the second features a half-and-half mix of computer and traditional work; the third moves on to work independent of the computers. Another course, Physics 5, is a five-quarter, in-depth version for physical science and engineering majors. Both courses are calculus based, stress problem solving, and cover the traditional ground of mechanics and the science of motion, electromagnetic theory, and the atomic and nuclear aspects of modern physics. Every 10-week quarter is divided into 8 units, each with computer dialogues and quizzes.



Professor Alfred Bork

Though it is only entering its ninth year as a formal program funded by the National Science Foundation, PCDP's roots actually go back much further. In 1958 Bork was teaching at the University of Alaska in Fairbanks when that school purchased a small, "primitive" computer. As a physicist, Bork was fascinated by it; as an educator, he quickly saw its teaching potential. He devised a small experiment with his physics classes in which computer programs reinforced classroom discussion.

Four years later Bork went to Harvard as an NSF science faculty fellow. Despite his Alaska experience, he had no idea how widely computers were used in instruction. He wrote to other colleges and to manufacturers. "Someone at IBM wrote back and said they didn't know too much about the subject," Bork recalls, "but they had heard that some teacher up in Alaska was experimenting with the idea."

Bork moved from Harvard to Reed College in Portland, Oregon, then after four years returned to Harvard to develop college computer materials and to work on a high school physics course. Meanwhile Irvine had been established as one of three new instant campuses in the growing UC system, with a heavy emphasis on computers. Through the influence of Ralph Gerard, first dean of graduate studies, "Computer U" had a computer even before it had a campus and was granted state legislative funds for experiments in computer-based teaching. Not surprisingly Bork was quickly recruited for the faculty. He started immediately to develop computer-based courses, but full adoption was delayed until the University changed computers in 1974. Today 57 percent of Irvine's students use computers during their four-year program, and Irvine ranks as one of the most computer-committed institutions in the country.

The current program is based on the Keller plan, or personalized system of instruction (PSI), but with unique refinements. Like PSI, the course allows students to study at their own pace and encourages them to review each unit until they demonstrate mastery. Bork delivers a weekly lecture and assigns a text – "Some students feel uncomfortable without a book or lecture," he says – but since the class may be involved in as many as eight different units midway through the course, neither print nor speech plays a prominent role. Instead, the student visits one of UCI's 25 graphic terminals in 3 locations, taps out a private code number, and works alone or in a small group. Studies show that the average student stays 25 minutes per sitting and totals about 2 hours a week.

In the computer room in Irvine's physical science building, students wait their turns at the Sigma 7 computer, the machine to which the physics courses are keyed. Those seated at the terminals watch as swift blips of light move slowly across the screen and form into the shapes and designs of physical phenomena. "To an outsider I guess it looks like we're playing Pong," says Eric Thompson, a senior biology major who has become so immersed in computer instruction that he now

helps prepare new dialogues. "But you really learn this way. Electric fields described in a book are difficult to visualize. When you can direct an electron through a field to a target you understand what it's about."

Although computer-based instruction may be thought of as passive and word oriented, Physics 3 is neither. Actual demonstrations appear on the screen and students manipulate the images so that they learn the principles by direct experience. Each terminal can project a set of cross hairs that can be moved by turning a knob on the keyboard. These enable the student to locate points on the screen or to indicate choices among phenomena. In an orientation dialogue about plotting coordinates, for instance, the student is asked to identify each point that indicates zero velocity. He simply moves the cross hairs to each point, punches the return, and the computer records his answer.

An example of how Physics 3 allows students to explore and reinforces their answers is demonstrated in the dialogue on magnetic fields, written by Bork and Arnold Arons of the University of Washington. First a simple circuit consisting of a dry cell, a light bulb, and a switch is sketched on the screen. A compass is placed beside the wire. The words "Now we close the switch" appear. The bulb lights; the compass needle shifts. "The lighted bulb indicates current," the computer prints. "Are you aware of what else happens in this system?" When the student replies that the compass points in a new direction, the computer immediately responds: "Yes. The compass points in a new direction. This physical effect was first observed by the Danish physicist Hans Christian Oersted in 1820. The discovery created a sensation and precipitated a line of investigation similar to the one we follow in this dialogue. Previously magnetic effects were known to be associated only with permanent magnets and not with electrical phenomena."

Next the computer shows in cross section a wire through which current is passing. The student is invited to use the cross hairs to situate a small compass in the region surrounding the wire. When he chooses his point and presses the F (for finished) key, the computer sketches in a compass with the needle properly positioned. Then the student is invited to choose additional locations for the compass. The computer adds more compasses and calls the student's attention to the pattern that has formed; when the student responds that he sees a counterclockwise circle, the computer congratulates him and moves on.

"We want to see more complete details of the program," the computer prints. "Consider your first compass location. We draw a line from this location to the wire. How would you describe the orientation of the compass needle with respect to the radial line?"

"It's perpendicular," the student responds. "Good," the computer enthuses. "The compass pointed at a right angle" – tracing the intersection of the lines – "to the radial direction. We can check this observation with another point. Point to one of your other compass locations" Manipulating the cross hairs, the student selects another perpendicular from wire to compass, then duplicates it at each location until the field appears as a pattern of spokes and the lesson is thoroughly reinforced.

To Bork, graphics are at the heart of his course; he believes his early attempts at computer-based instruction were less successful until the University acquired a computer with greater graphic capability. "Whether or not a picture is worth a thousand words," he says, "there is no question that often we can get an idea through a picture in a way that is either an alternative to verbal and numerical ways of approaching it, or a view of the information that is simply not available in other ways."

The dialogues also employ typographical pyrotechnics: Varied letter styles, underscores, boldface, and capitals catch student interest and make emphases. Flexibility is also important. The computer will accept most variations on the correct answer and respond accordingly. It also is programmed to reply to the most common of incorrect answers, and after a pattern of mistakes will direct the student back for further study. The responses are always enthusiastic and supportive. "First-rate!" may be the comment after a correct answer. Or "Right on!" At one point, the student is told, "Relax."

Each student takes eight or more quizzes per quarter – at least one per unit – and may decide for himself when he is ready for them. The quizzes are taken and graded immediately "on line" via a pass/repeat system. The student is given 40 minutes to complete the questions. If he does not achieve a satisfactory grade, he may try again the following day. He is allowed four chances per quiz before the computer refuses further testing and refers him to the instructor for help. With its flexible features, the system can offer each student a unique examination.

"Computer U" had a computer even before it had a campus and was granted state legislative funds for experiments in computer-based teaching.

The system's storage bank also allows the student – and the instructor – continuous feedback on progress. By punching the student's code number either may obtain an up-to-the-minute report on the number of dialogues completed and tests passed. The student also is told his current approximate grade, based on points for each completed dialogue and quiz, plus bonuses for completing them before a deadline. Bork also can learn how often the student visits the computer and how much time he spends there, so that he can step in when a student is procrastinating.

PCDP has made a strong effort to avoid a pitfall of PSI; since it functions on self-discipline, it tempts students to delay until they are too far behind to catch up. At the first class, Bork warns of these problems and urges students to map a course strategy. Notes the course guide: "THE GREATEST PROBLEMS with PSI courses occur

for students who do not construct and carry through such plans for themselves, but only carry out a particular week's work under pressure." Bork suggests that students either set and maintain a standard pace, move ahead of the pace and use the time for other activities, or plan their physics study around other courses.

"We use the carrot and the stick," Bork says. "The carrot is in the form of bonus points for completing quizzes before an established deadline. The stick consists of a rule that you cannot complete more than four units in the last three weeks." The aides and the instructor are on call daily, and students needing special assistance can leave messages for Bork in the computer.

Overall, students praise the system. "It's much more of a learning experience than either a textbook or a lecture," says Sylvia Noronha, a senior biology major. "You become almost mesmerized by it." But the students are often annoyed by technical problems. "Sometimes you have to wait two hours for a terminal," complains Stephen Simon, a premed student. "And the computer breaks down an average of once a week." Even waiting in line doesn't guarantee access to the computer. "The system is so overloaded that it's a real lottery," says Eric Thompson. "When you get a terminal you have to wait until somebody logs off, and then you come on simply because you have started at that same split second." Terminals are available to students from 6 a.m. to 1 a.m., but even at 11 p.m. lines sometimes exist, especially at quarter's end. Commuters gripe that they spend long hours at the campus simply waiting for a terminal.

The Physics Computer Development Project has received national attention, but it has been less successful in winning support of other members of the physics department. William Parker, who also teaches introductory physics, sticks to lectures. "We have to provide a variety of mechanisms for learning, but I look to the computer to supplement the lecture, not replace it. It cannot provide some of the subtleties, some of the sophistication about the subject that a student can obtain from the human presence." This winter Parker and Bork will team teach Physics 5; Parker will lecture, and Bork will use computer-based instruction.

Online at Irvine's Physics Computer Development Project.

Although such studies have not been conducted at Irvine, Bork quotes research elsewhere to indicate that computer-based instruction costs only two thirds as much as conventional teaching and that students of matched ability do equally well whichever system is used. His own experience indicates the method is particularly effective for the average student, because he is able to review the material many times until he has mastered it. Bork acknowledges some of the technical problems – students complain that typing errors often compel them to repeat dialogues or even yield a wrong answer – but says these bugs are being worked on.

Michael Scriven, professor of philosophy at Berkeley, last spring conducted a formative evaluation of the PCDP. Relying on interviews with students and using feedback from the computer records, Scriven suggested a battery of improvements, including rewriting certain dialogues to improve clarity and revising the point system so that students were no longer penalized because of delays in reaching a terminal. In accordance with Scriven's recommendations the written final has been made mandatory in order for a student to receive an A; B has become the highest grade obtainable on the basis of online quizzes alone.

Meanwhile, the ripple effect has carried Bork's influence beyond the physics department. A computer-based introductory mathematics course has been inaugurated at Irvine, and computer-based units are included in courses in chemistry, ecology, anthropology, and medicine. The physics professor believes that the spread will become even greater with wide adoption of minicomputers. The present cost for computer equipment is $300 per student, he says. Newly available technology can reduce it to $30, and with current home computers that operate without sophisticated graphics, it comes down to $3 per student. The economics of it all someday will make computer-based instruction irrestible.

For many faculty, including physics faculty, that is a day to be wished for, Bork says. "There will be many benefits from using the computer as a learning device. Routine activities like record-keeping will be left to the computer. Faculty can spend more time working individually with students. Surely this will seem more rewarding and exciting than lecturing to a crowd of 500. Used properly, the computer can lead to a more humanistic approach to education."

This report, made possible by the Fund for the Improvement of Postsecondary Education, appeared in Change Magazine*'s "Report on Teaching", Jan. 1978.*

FACTS ON FLOPPIES

CHIP A. TYETI

If you're developing a healthy library of programs for your micro or have extensive files and records to maintain, a floppy disk operating system might be the perfect adjunct.

You'll want to consider several criteria before buying your FDOS: the size diskette used, whether the unit is a kit or (partially) assembled, whether it is hard or soft sectored, availability and other factors. In flexibility, floppy disks resemble thin sample records and promotional records bound into the pages of magazines. The surface has an ultrasmooth magnetic quality unlike the visible grooves in a standard audio record. Your disk should remain inside its sturdy paper envelope so it does not become bent or damaged.

Manufacturers offer two sizes of floppy diskettes and disk drive units. The smaller, 5-1/4", is more compact, but consumes more time if you have large files to maintain – you might be changing diskettes more frequently than you planned. The larger diskettes, 8", give you considerably more bytes per disk, but require more space. This will make a difference if you want a relatively portable system.

Some manufacturers offer single and/or double density in both sizes. Double density, as its name suggests, compresses more bytes per sector, but then becomes incompatible with the IBM 3702 format (128-byte sectors). Thus, you cannot exchange a diskette from a double density drive to single density drive. The same problem applies to interchanging hard and soft sectored diskettes.

Diskettes contain several concentric tracks divided into sectors (like pieces of pie). The 5-1/4" diskette contains about 35 tracks, while the larger 8" diskette contains about 77. Both diskettes are divided into about 26 sectors.

To learn the capacity of your (proposed) system, multiply the number of sectors by the number of tracks by the number of bytes per sector by the density of the drive by the number of sides available per diskette.

A hard sectored disk has sectors delineated by guide holes punched into the surface of the diskette. These holes help in positioning the head for reading/writing. The soft sectored diskettes have just one hole for the initial alignment and are software-controlled for the remaining sectoring of the diskette.

For personal applications, research and comparatively small business applications, the floppy concept is ideal. If you need more storage and quicker access, rigid disk systems may prove more efficient.

The rigid disk contoller PC card manufactured by CreaComp Systems is one example of a comparative unit which can be made compatible with almost any micro/mini system. Cost of this card is nominal compared to the expense of rigid disk drives. And watch this spring for a complete package from CreaComp that might really wet your whistle.

Micropolis of Canoga Park, CA, offers a 5-1/4" floppy with the same capacity as an 8" unit. MetaFloppy, as they call it, uses 77 tracks, each with 16 sectors of 256 bytes per sector, yielding a capacity of 315K bytes per drive! If you plan to use disks only on your own system or in exchange with another Micropolis system, then the compactness and efficiency of their system is perfect.

Remember – hardware, software and supplies to maintain an FDOS are generally incompatible. You will need both software and hardware to bring up your system in a specific configuration.

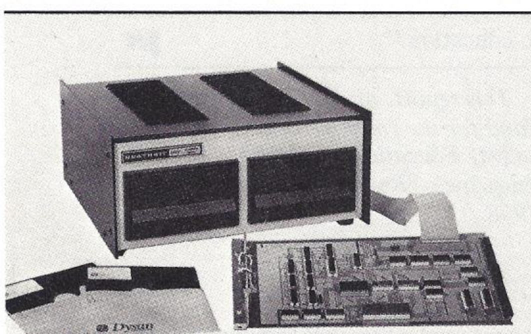
Price difference for a kit or an assembled unit generally refers to the status of the controller PC card. Drives come completely assembled (except for power supply, cabinet and cables).

All prices listed in our chart refer to a dual drive, single density, single-sided unit with all the necessary hardware and software to run immediately. This allows for a back-up drive and any additional costs before linking the FDOS with your own system. Please read the extensive footnotes carefully.

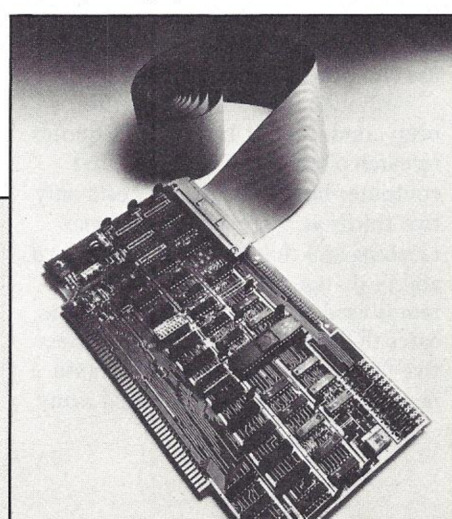
Some disk controller cards control up to eight disks. Cromemco's board, for example, handles a combination of both large and small disk drives. Consider this feature if you must maintain extensive (inventory) files, yet your basic programs are moderately small. Or maybe you want to keep essential data on one type of drive and dump out to the other drive data that could be exchanged into another system (submit your IRS tax return in IBM 3702 format only; keep your journal and ledger accounting data on the other drives). PerSci expects to offer a controller in June that will handle up to 16 drives. Fancy that!

Keep in mind that old, and ofttimes aggravating, availability factor. Problems develop for both the manufacturer and the end user when companies hold out on delivery because of incom-

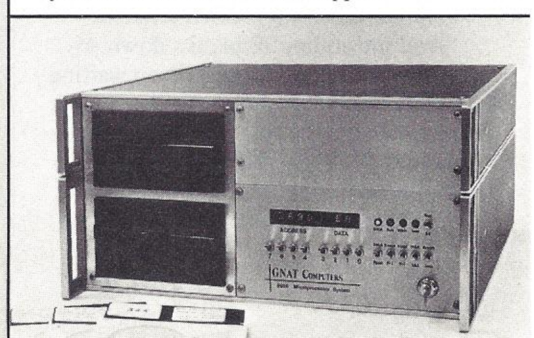
The H17 is Heath's disk mass storage peripheral designed for the H8 computer.



Vista's S100 controller card for their floppy disk operating system.



GNAT integrated floppy microcomputer system also available as minifloppies.



FLOPPY FACTS

Mail Order/Distributor Retailers/Manufac. Reps	Warranty Period	Manufacturer	Size of Diskette in Inches	Kit/Assembled	Price[21]	Hard or Soft Sectored[23]	Density[24]	Max Diskettes Controlled	Single or Double Sided	Availability[25]	Maximum Capacity of (Dual Drive) System	Bus Structure	Amt. of Mem'y Required for Operating System	Disk Drive OEM	Minimum Amt. of Memory Required for Their BASIC	Does the Operating System Contain an Assembler?	Does the Operating System Contain a FORTRAN?	Does the Operating System Contain a BASIC?
RE	1y	Apple Computer	5¼	A	$[1]	S	S	3	S	3	226K	Apple	[1]	SH	16K	Y	N	Y
MR	120d	CalComp	8	A	3630	B	B	4	B	1	2M	[2]	2K	CC	[1]	N	N	N
RE	1y	CreaComp Systems	H	A	[1]	[1]	[1]	4	D	3	12M	S100	[1]	[1]	[1]	[1]	[1]	[1]
RE	90d	Cromemco	8	K	2485	S	S	4	S	3	500K	S100	5¼K	PS	32K	A	[1]	A
RE	90d	Cromemco	8	A	3185	S	S	4	S	3	500K	S100	5¼K	PS	32K	A	[1]	A
RE	90d	Cromemco	5¼	K	1480	S	S	4	S	1	92K	S100	5¼K	WN	32K	A	[1]	A
RE	90d	Cromemco	5¼	A	1680	S	S	4	S	1	92K	S100	5¼K	WN	32K	A	[1]	A
RE	90d	Digital Group	5¼	K	1195	S	D	4	S	2	160K	DG	2K	PT	16K	F	[1]	A
RE	90d	Digital Group	5¼	A	1395	S	D	4	S	2	160K	DG	2K	PT	16K	F	[1]	A
RE	90d	Digital Group	8	K	1575	S	S	4	S	1	512K	DG	2K[3]	SH	16K	F	[1]	A
RE	90d	Digital Group	8	A	1795	S	S	4	S	1	512K	DG	2K	SH	16K	F	[1]	A
RE	90d	Digital Systems	8	A	2545	S	B	8	S	1	512K	S100	6K	SH	18K	Y	A	A
MO	90d	Ebnek	8	A	2650	S	S	4	S	3	512K	72 line[4]	4K	PS	[1]	[1]	[1]	[1]
MR	90d	Elec'c Prod't Assoc.	8	A	3295	S	S	4	S	1	1M	[5]	8K	PT	16K	Y	A	A[6]
RE	90d	Extensys[7]	8	A	3995[8]	S	S	4	D	2	4M	S100	8K	SH	[1]	Y	A	A
MO	1y	General Robotics	8	A	10950[9]	S	D	4	S[10]	1	1.24M	QBUS	8K	PT	12K	Y	A	A
MR	1y	GNAT	8	A	5500	S	S	4	D	1	500K	GNAT	16K	SH	32K	Y	A	A
MR	1y	GNAT	5¼	A	3690	S	F	3	F	1	160K	GNAT	16K	SH	32K	Y	A	A
RE	[1]	Heath H17 for H8	5¼	A	1000	H	[1]	[1]	[1]	3	204K	H8	15K	WN	[1]	[1]	[1]	[1]
RE	[1]	Heath H27 for H11	5¼	A	2150	H	[1]	[1]	[1]	3	204K	H11	15K	WN	[1]	[1]	[1]	[1]
MR	[1]	ICOM FD3712	8	A	[1]	S	S	4	S	1	~.5M	[11]						
MR		ICOM	8															
MR		ICOM	5¼															
RE	90d	IMSAI	8	A	3164	S	S	4	S	3	~.5M	S100	16K	CC	20K	Y	A	Y
RE	90d	IMSAI	8	A	2145	S	B	7	S	3	~.5M	S100	16K	PS	20K	Y	A	Y
RE	90d	IMSAI	8	A	2964	S	S	4	S	3	~.5M	S100	16K	CC	20K	Y	A	Y
RE	90d	IMSAI	5¼	A[12]	1290	S	S	3	S	2	180K	S100	16K	SH	20K	Y	A	Y
RE	90d	Info 2000	8	A	2650	S	S	4	F	1	512K	[13]	[14]	PS	16K	A	A	A
RE	90d	Micromation	8	A	2195	S	S	8	S	3	512K	S100	-5K	PS	24K	Y	A	Y
RE	6m	Micromation	8	A	1895	S	F	8	F	1	512K	S100	-5K	MM	24K	Y	A	Y
RE	90d	Micropolis	5¼	A	1895	H	Q	4	S	1	630K	S100	24K	MC	136K	Y	N	Y
		Micro Systems Dev.	5¼	K														
		Micro Systems Dev.	5¼	A														
RE	90d	Midwest Scientific	8	K	2065	H	S	4	S	1	630K	SS50	16K	GSI	24K	Y	N	A
RE	90d	Midwest Scientific	8	A	2410	H	S	4	S	1	630K	SS50	16K	GSI	24K	Y	N	A
RE	90d	North Star	5¼	K	1255	H	S	3	S	1	180K	S100	16K	SH	16K	N	F	Y
RE	90d	North Star	5¼	A	1425	H	S	3	S	1	180K	S100	16K	SH	16K	N	F	Y
RE	1y	Ohio Scientific	8	A[17]	1590	H	S	2	S	2	512K	OSI	13K	GSI	8K	Y	N	Y
RE	90d	Peripheral Vision	8	K	1693	H	S	8	S	2	606K	S100	10K	[15]	16K	Y	A	Y
RE	90d	Peripheral Vision	8	A	1818	H	S	8	S	2	606K	S100	10K	[15]	16K	Y	A	Y
RE	90d	PerSci 2042	8	A	3120	S	B	4	S[16]	3	506K	[13]	-4K	PS	[1]	[1]	[1]	[1]
RE	90d	PerTec/MITS																
RE	1y	Processor Technology	8	A	2895	H	S	8	S	1	3M	S100	32K	PS	16K	Y	A	Y

ALL FOOTNOTES ON NEXT PAGE

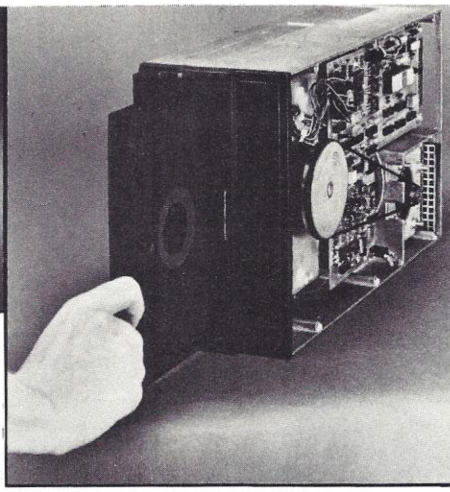
FLOPPY FACTS

Mail Order/Distributor Retailers/Manufac. Reps	Warranty Period	Manufacturer	Size of Diskette in Inches	Kit/Assembled	Price[21]	Hard or Soft Sectored[23]	Density[24]	Max Diskettes Controlled	Single or Double Sided	Availability[25]	Maximum Capacity of (Dual Drive) System	Bus Structure	Amt. of Mem'y Required for Operating System	Disk Drive OEM	Minimum Amt. of Memory Required for Their BASIC	Does the Operating System Contain an Assembler?	Does the Operating System Contain a FORTRAN?	Does the Operating System Contain a BASIC?
RE	90d	Radio Shack (Tandy)	5¼	A	1297	S	S	4	S	1	142K	RS	16K	SH	16K	F	F	[1]
RE		Realistic Controls	5¼	K														
RE		Realistic Controls	5¼	A														
		Seals																
RE	90d	Smoke Signal Bdcstng.	5¼	A	1009[18]	S	S	3	S	1	160K	SS50	4K	SH[19]	16K	A	N	Y
RE	90d	Southwest Tech Prod.	5¼															
DI	90d	Sykes Datatronics	8	K	2680	S	B[20]	4	S	2	512K	[11]	[1]	SY	[1]	[1]	[1]	[1]
DI	90d	Sykes Datatronics	8	A	3980	S	B[20]	4	S	2	512K	[11]	[1]	SY	[1]	[1]	[1]	[1]
RE	90d	Synetic Designs	8	A	2495	S	S	4	S	1	512K	S100	12K	PT	20K	Y	N	A
RE	6m	Tarbell Designs	8	K	1480	S	S	4	S	2	512K	S100	6K	[22]	20K	Y	A	Y
RE	6m	Tarbell	8	A	1570	S	S	4	S	2	512K	S100	6K	[22]	20K	Y	A	Y
RE	6m	Technico	8	A	2395	S	B	4	B	2	~1M	TI9900	1K	SH	12K	Y	F	A
RE	90d	TEI	8	A	3275	S	B	8	B	2	512K	S100	[1]	SH	20K	Y	A	Y
RE	90d	Vector Graphics	8	A	2300	S	F	4	S	1	512K	S100	16K	PS	24K	Y	A	Y
RE	90d	Vista	5¼	K	1204	S	F	8	F	1	160K	S100	16K	SH	24K	Y	[1]	Y
RE	90d	Vista	5¼	A	1374	S	F	8	F	1	160K	S100	16K	SH	24K	Y	[1]	Y

FOOTNOTES

1 contact manufacturer for specific configuration data
2 compatible with RS232C, LSI-11, and S100 bus structures
3 available soon: a 6-8K operating system which will be more sophisticated
4 for TMS 9900
5 DG and S100 busses as well as Motorola 86-pin (exorcisor) bus
6 this is a compiler
7 the interface card contains an 8080 μp, 1K ROM, 8K RAM (freeing up 8K of system RAM)
8 double-sided price ... ...
9 includes all but the terminal
10 available in double-sided format during second quarter of 1978
11 may be made compatible with almost any μp
12 H8, S100, and DG busses
13 compatible with 8080 interface
14 includes 3K monitor, 1K scratchpad on controller card; no system memory req uired
15 originally Innovex; now supplied by Shugart
16 a new controller available about May, 1978, will handle up to 16 diskettes
17 also available in kit form
18 depending on how the specific order was written, the price is either \$1009 or \$1139
19 some drives are from Shugart, others from MPI
20 when double density is operational, a hard sectored diskette must be used
21 standard used: single-sided, single density dual drive unit with all the necessary hardware and software to run the system immediately
22 some drives are from PerSci, others from GSI
23 B=both hard and soft sectored; F=to be offered in the future
24 S=single; D=double; B=both; F=to be offered in the future
25 1=within one month; 2=between one and two months; 3=beyond two months

Peripheral Vision's 303K bytes of formatted data capacity.



Calcomp's quad-density disk drive with advanced button head design.

Micropolis 1053 Mod II has 630K bytes of on-line storage.

plete products, when they accept orders in advance of release dates, when order fulfillment takes weeks or months or when particular units are scrapped for something due to come out in six months.

Several manufacturers noted that upgrading single density drives to double density would be relatively minor, but the components may not be available for months. Other manufacturers mentioned that single-sided drives cannot be upgraded to double-sided drives because the read/write heads must be on both sides. Thus, you'll have to acquire replacement drives to achieve double-sided disk reading/writing.

The industry moves with such speed that remarkable changes in the entire FDOS concept may be on the drawing boards now, but not breaking news for a year or more. You must consider such factors in a logical plan for both short and long range expectations.

As you scan the "maximum capacity of the system" column, you'll see an extremely wide range of storage capacities. Some vary with cost; others relate to hardware sophistication.

We included variable bus structure listings for Digital Group buses, Heath buses, SS50 buses, and S100 buses in addition to several other bus structures. Some companies said that their controller card can be made compatible with any 8080-based system.

Note that some manufacturers listed sell their drives (with or without controllers and compatible software) in "quantity one" to end users, although their primary market is systems manufacturers who interface their drives into a complete package. These disk manufacturers add one more step to the manufacturer-to-owner chain and thus add potential for more problems. If problems develop, consult your local dealer or rep, the systems manufacturer, the OEM source, or finally, with your psychiatrist. No one ever said life with a microcomputer floppy disk operating system was going to be pleasant – but most manufacturers and dealers are courteous, knowledgeable and usually more than willing to lend a helping hand. Make sure you distinguish between paying for the service and chatting on friendly (or not so friendly) terms.

Our chart also lists amount of memory required for each operating system. Responses from Micromation and PerSci indicated that on-board PROMs provide the necessary operating system, giving a negative value for the amount of memory required. We also list the amount of memory required for the supplied BASIC.

The final phase of the chart delineates which FDOSs contain assemblers, FORTRAN, and BASIC. If these software packages are not included in the price listed, the designation "A" notes their availability at additional cost.

There has been a great deal of conversation – perhaps even debate – on the pluses of using CP/M (a software package of Level I language used to control your disk drives) as part of the operating system. CP/M, originally written for the IBM format, is now available for North Star systems for \$112 from Lifeboat Associates. Note that IBM format is soft sectored while North Star is hard sectored. Before this offer, Keith Parsons of Structured Systems Group, another software publisher, suggested that if you wanted CP/M for your North Star system, you would have to pay more than \$1500 for encoding . . .

Realistic Controls of Davenport, IA, offers FORTRAN with their FDOS, but according to Jim Nance of the Computer Store of Davenport, any Sol owner wishing to hook up the RC system must either cut all the pull-down resistors on the data bus or replace them with resistors 10X the original resistance. This example shows the kind of compromise you may need to make to get an FDOS running with your micro.

Vista, a division of Randal Data Systems, offers software functions including instantaneous program loading, named dynamic files, program editing, debugging, assembling, batch processing and file copying on back-up diskettes. Vista offers the only minifloppy now in the marketplace with soft-sectored CP/M ready to run in the drives.

Many manufacturers offer FDOSs as an integral part of their microcomputer system. With our chart of peripherals we're showing you what's available on an individual basis.

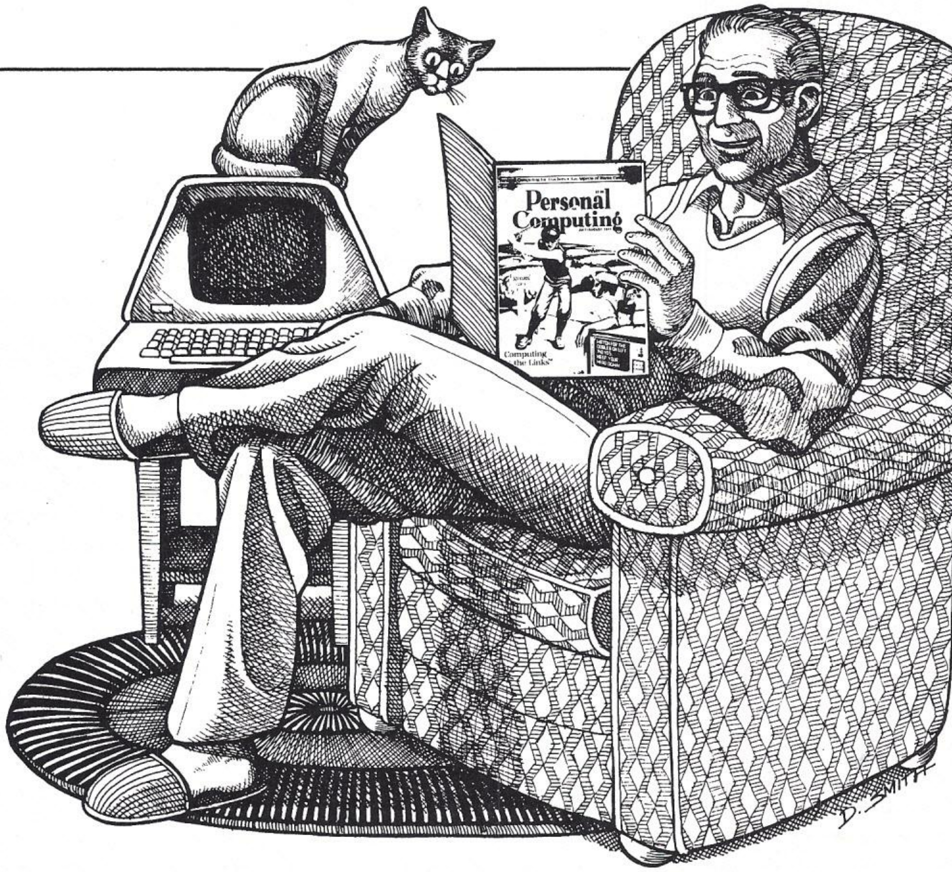
Heath H17, a floppy disk mass storage peripheral designed exclusively for the H8 computer, will be introduced in a fully assembled form sometime in June, 1978. The H17 uses Wangco's 5-1/4" drives. Initially you must order a second drive for the cabinet, because the H17 will not come as a dual drive.

Software included with the H17 consists of the Heath Disk Operating System (HDOS), HASL-8 assembler, TED-8 editor, BUG-8 debugger and extended Benton Harbor BASIC. Kit versions will follow.

Micropolis units offer several extras worth noting: because you buy the drive from the manufacturer, you avoid another middleman. Micropolis is the only system builder completely integrated in manufacturing. A phase locked loop technique integrated into their data separator assures data reliability. Micropolis units include automatic deselection feature which relieves head pressure on the recording surface when the disk isn't in use.

Once you make a thorough evaluation of your finances, spare time (for kit building), record keeping needs and overall aesthetic desires, visit your local computer store to discuss your plans. Talk to friends in computer clubs regarding the interfacing requirements of your system. Spend a few dollars (before you waste a few hundred) by calling the companies. Discuss your intentions with a customer service engineer – not necessarily a sales representative.

A well-organized plan for evaluating units you've selected as "possibles" will save you time, money and needless frustration.



Leading the way to computer mastery . . .

Maybe you own a computer. Maybe you're thinking about buying one. Or perhaps you're just curious about the computer world. Let *Personal Computing* show you how to use microcomputers for fun and profit.

A year's subscription brings you 12 monthly issues filled with informative, easy-to-read articles on all aspects of personal computing. Games to play, money-making ideas and applications and programs for home, business and educational use are just a few of the topics covered in *Personal Computing.*

We provide instructive articles on computer basics as well as more advanced articles.

So learn to master your computer the easy way; curl up in your favorite chair with *Personal Computing.* Use the attached coupon to order your subscription today.

Personal Computing Magazine

I've been bitten by the bug. Enter my subscription to Personal Computing today!

NAME____________________________________

ADDRESS__________________________________

CITY__________________ STATE______ ZIP______

USA

☐ 1 year (12 issues) $14
☐ 2 years (24 issues) $26
☐ 3 years (36 issues) $38

☐ Bill me
☐ Check enclosed
(you'll receive one extra issue for each year!)

Charge my:
☐ Master Charge
☐ Bank Americard

Account #____________
Card expiration date________

ADDITIONAL POSTAGE (per year)

CANADA & MEXICO	$4.00 surface	$8.00 air
OTHER FOREIGN	$8.00 surface	$36.00 air

(Please remit in US funds – Thank you)

BACK ISSUES $3.00/COPY
(Payment must accompany order.)

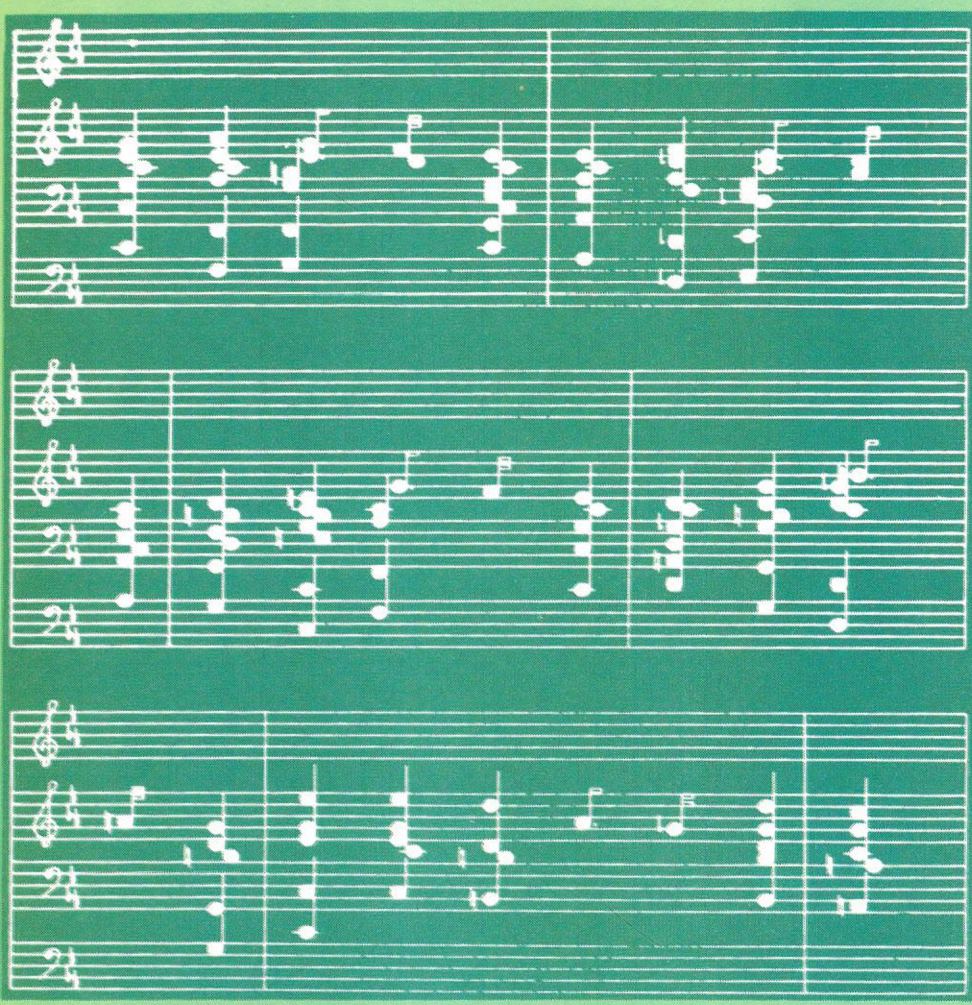
Mail to: PERSONAL COMPUTING, 1050 Commonwealth Ave., Boston, MA 02215

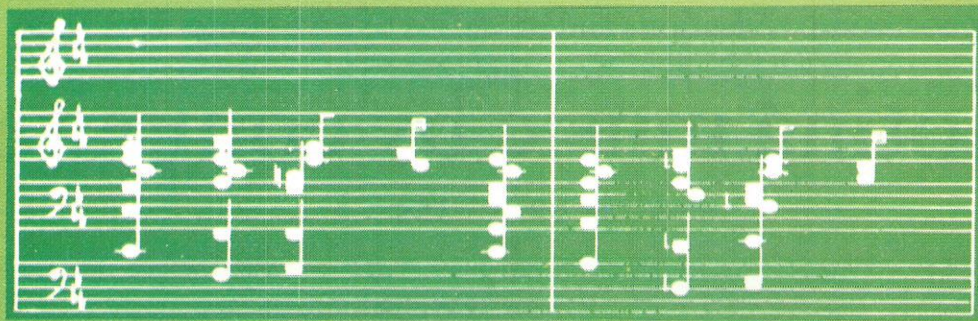
A system that automatically transcribes music played on keyboard instruments, records improvisations and promises to save composers' time.

Noting music by computer

BY DR. P. MARS

Reproducing a musical score automatically is not a new idea. The first version of an automated piano, in the form of a pianista, was introduced more than a hundred years ago. Later developments include the well-known pianola and the piano camera.* Our own research stemmed from an interest in modern jazz piano.

There is a tremendous scarcity of original, accurate transcriptions of such music. Many jazz pianists have had no classical training in music, and even those who have cannot spare the time for the tedious work of transcribing. But the keyboards of the piano and organ are ideally suited to the computer. It is a reasonably simple matter to arrange that every time a note is struck it is recorded, and to monitor accurately the times when they are struck. All the information can be recorded in the binary form of 1 and 0, that is, the language of the digital computer. Information can be monitored during a performance by using an automatic transcription unit and storing it in digital form on a cassette recorder. It can then be processed by computer to produce a transcript, in musical notation, of the original keyboard performance.

Transcription of Chopin's *Prelude in C minor,* opus 28, No. 20.



The transcription unit samples the entire keyboard at a rate of, for example, 20 times/second throughout the performance and the information on pitch and timing of notes, after some manipulation, is recorded. No audio frequency needs to be recorded; all that is wanted is digital information, so it does not matter if the piano is out of tune or even if a dummy keyboard is used.

It is quite simple to connect the automatic transcription unit and cassette recorder to any keyboard device, but although direct electrical connections can be made to electronic organs and pianos, optical transducers are needed to convert the key movements of ordinary pianos.

During a performance any notes struck by mistake are, of course, transcribed, for the print-out is not governed by any law of musical tonality. Difficulty might arise in drawing the lines between the bars of the music because the musician seldom sticks exactly to a strict enough tempo to follow a particular crotchets/minute count, so the bar lines may be incorrectly placed. However, if the performer does stay within the constraint and tolerance of a specific count, the computer can draw bar lines quite simply. Unfortunately, for some practical applications such as transcribing *avant garde* jazz, timing within a piece modulates and may have random variations.

A further disadvantage of the system is that it offers little or no discrimination between which hand plays which note or set of notes. All note tails are drawn upwards and no distinction is made between lower and upper hand in the print-out. For similar reasons, no rests are drawn; it is impossible, for a particular piece, to ascertain individual voicings. Rests must be added by the composer after the automatic transcription has been made. No expression marks are incorporated automatically, either, because modelling musical expression mathematically poses an unsolved problem; all expression marks must be added later by the composer. The system allows a key to be specified but many compositions involve changes in key and it is not practical to account for them 'on-line', during performance. This information must be added later, 'off-line'.

In spite of these limitations, the machine gives a completely literal transcription in terms of note pitch and time, making the system attractive as a potential labour-saving device for musicians.

We intend to add to the system in the near future, to permit the original transcription to be edited with the aid of a conventional visual display unit. The composer will be able to insert expression marks, rests and so on automatically.

Fast

Recent work in conjunction with the well-known jazz pianist Oscar Peterson has shown that the transcription system can cope with the fastest of jazz improvisations. It is a relatively simple matter to play back original transcriptions under remote computer control, and thereby provide an audio check on their validity. It is also possible to include semi-automatic composition. For example, given a standard popular tune, the computer can be organized to play the standard left-hand chord sequence and generate jazz improvisation, superimposed on the original chord sequences. For any chord, notes that obey the standard harmonic laws can be randomly selected for improvisation. Every improvisation so produced is original and the composer can simply select the most attractive; the automatic transcription system then produces a conventional music-notation output.

Although the system was originally developed to solve problems associated with jazz piano, it can obviously be applied to all forms of keyboard music.

---

* Further information may be obtained from the Musical Museum, 368 High Street, Brentford, Middlesex TW8 0BD.

Reprinted by permission. Robert Gordon's Institute of Technology, Aberdeen. No. 151/1971/ Spectrum/7

IT'S A GREAT BIG COMPUTER WORLD
But You Only Need

THE COMPUTER CORNER

- SOL — A New Dawn Is Here!
- IMSAI 8080
- POLY — 88
- TDL Z-80
- Memories & I/O Boards

- Computer Book Service
- Magnetic Tapes & Disks
- Full Line of Magazines
- Brain Games & Puzzles
- Workshops & Club Information

Visit **THE COMPUTER CORNER** for all your computer needs. Stop in and browse — you'll like our personal service.

THE COMPUTER CORNER
White Plains Mall — Upper Level
200 Hamilton Avenue
White Plains, New York 10601

Tel: (914) WHY - DATA

Ample Parking
10-6 Daily & Saturday
10-9 Thursday

CIRCLE 21

INQUIRE TODAY!!

We Ship From Stock

Business Systems Are Here Now!

COST EFFECTIVE HARDWARE • PERIPHERALS • SOFTWARE

Call our Mr. Amunson

Informative assistance in defining your objectives and choices.

For Engineers and Serious Hobbyists

BREADBOARDING • WIRE WRAP • DESIGN EQUIPMENT • TEST EQUIPMENT • COMPONENT PARTS • POWER SUPPLIES

• AP Products • OK Machine • Iso-Tip • B&K PRECISION • CONTINENTAL SPECIALTIES • NON-LINEAR SYSTEMS • PANVISE • VECTOR

Our 25th Year

ATLAS ELECTRONICS CORP

1570 3rd AVE. (88th St.) NEW YORK, N.Y. 10028
212-427-4040

Send $1 for complete set of catalogs (refunded on 1st purchase)

**We ship nationwide & overseas*

CIRCLE 22

!EXTRA!

High Prices Byte The Dust.

(Fayetteville, NY) Computer Enterprises has kept their prices low. If you order now, on top of reliability, lightning-fast delivery, you'll pay only the prices you see below. Other great products from the leaders in micro-computing are available from Computer Enterprises.

	Credit Card Price	Cash Price
Lear Siegler ADM-3A Kit	$728	**$700**
IMSAI 8080 w/22 slots Kit	614	**590**
IMSAI PCS-80/30 Kit	1097	**1055**
IMSAI PCS-80/15 Kit ...	731	**703**
Edge Connectors & Guides for IMSAI	4	**4**
Heuristics Speechlab Assmbld.	265	**254**
Vector Graphic 250ns 8K RAM Assmbld.	187	**180**
TDL VDB Video Display Board Assmbld.	338	**325**
TDL Software Package A on North Star Disk	228	**219**
TDL Software Package on Paper Tape	150	**144**
George Risk ASC II Keyboard Kit	52	**50**
Cromemco Dazzler Kit	190	**183**
Cromemco 16K PROM Card Kit	128	**123**
Cromemco TU-ART Kit	173	**166**

Call or write for our giant Spring Computerlogue Catalogue Today.

IMSAI VIO-B Kit	252	**242**
IMSAI VIO-C Kit	297	**286**
TDL XITAN ALPHA 1.5	795	**764**
Cromemco Z2 Kit	557	**536**
Dynabyte 250ns 16K Static RAM Assmbld. ..	577	**555**

Shipping charges: $10 per CPU on larger units; $1.50 per kit. $2.00 min. per order.
Delivery is stock to 30 days on most items. Shipment is immediate for payment by cashier's check, money order or charge card. Allow 3 weeks for personal checks to clear. N.Y. State residents add approp. sales tax. Availability, prices and specs may change without notice.

Operating Hours:
M-W 10-5 E.S.T.
Th-F 10-9 E.S.T.
Closed Sat. & Sun.

Write or Call

computer enterprises™

P.O. Box 71
Fayetteville, N.Y. 13066
Phone **(315) 637-6208** Today!

CIRCLE 23

Volumes VI & VII are finally released!

Volume VI

Yes and it still contains what was previously advertised. A fully disk interactive business package with A/R, Inv., A/P, ledgers, tax totals, payroll records, more.

As a bonus it also contains the Users Manual for our Firmware Ledger package. These 100 extra pages contain report formats, file creation routines and our very powerful program ACBS1 used to create the powerful file structured data base.

$49.95

Volume VII

Here is that Chess program you have been waiting for as well as a disk interactive Medical Billing package with patient history file.

Also included is our disk interactive Word Processing package (revision Ø).

$39.95

Add $1.50/Vol. for U.P.S. and handling except to APO and PO addresses. Foreign orders add $8/Vol. for air shipment — US dollars only. No purchase orders over $50.

VOLUME I —	$24.95
VOLUME II —	24.95
VOLUME III —	39.95
VOLUME IV —	9.95
VOLUME V —	9.95

OUR SOFTWARE IS COPYRIGHTED AND MAY NOT BE REPRODUCED OR SOLD.

Due to the numerous copyright violations on our earlier volumes — until further notice we are offering a REWARD leading to the arrest and conviction of anyone reproducing our software in ANY way without our written permission. This includes diskettes, paper and magnetic tape, cassettes, records, paper copies, etc.

SRI SCIENTIFIC RESEARCH

220P Knollwood
Key Biscayne, Florida 33149

Phone orders 800-327-6543
305-361-1153

AVAILABLE AT MOST COMPUTER STORES

BankAmericard / master charge honored

Product Ponderings

BY RUSS WALTER

Personal computers vary in style, capability and price ranging from inexpensive pre-programmed game systems to the more advanced, and expensive, home control systems. For under $25, you can buy an entire pre-programmed game computer (including the CPU, memory, input/output devices, software and extras) that fits in the palm of your hand and plays an Auto Race on its video screen. You can get it from Mattel, along with two other games: Missile Attack (under $25) and Football ($30 - $35). Each pre-programmed game comes in its own self-contained package.

With the Mattel product you cannot create your own program but must use the one they supply. Other companies, such as Fairchild, sell video games that attach to your television.

Other types of such "dedicated applications" (programs designed for one specific function) include car fuel emission control, microwave oven timers, burglar alarm systems and more.

Development systems offer another alternative for the personal computerist. Among hobbyists, the most popular development system is the KIM-1, made by MOS Technology. For $245, you get a CPU, 1K RAM, 2K ROM (containing an operating system), hexadecimal keyboard, LED read-out and interfacing for an optional terminal and for an optional cassette tape recorder.

The entire computer comes completely assembled and ready to run, except for the power supply (which costs an extra $25). You'll spend much more money than if you buy a pre-programmed video game, but you gain the flexibility of being able to invent your own programs. Unfortunately, you must write the programs in machine language, unless you want to spend extra money for BASIC, which in turn costs more for a terminal, an audio tape cassette and more RAM memory. By the time you buy these extras, you'll discover you would have been better off buying a consumer system . . .

Consumer systems manufactured by both Commodore and Radio Shack offer such good deals that they're scaring the rest of the computer industry. For "about $600", you get a CPU, terminal, cassette tape recorder, ROM (containing the operating system, editor, graphics package and BASIC), and 4K RAM completely assembled (including interfacing and power supplies) and ready to run.

But the "about $600" figure has a catch. For Radio Shack, the price includes a poor version of BASIC (called "Level 1"); for better BASIC, you must pay a surcharge. For Commodore, the delivery will be slow, since production of the 4K RAM version has been delayed; if you want to get your computer relatively quickly, you'll have to buy the 8K RAM version, for which Commodore charges an extra $200.

I recommend Radio Shack over Commodore, because Radio Shack's terminal has a larger screen, the keyboard is easier to use, additional memory costs are less (increasing from 4K RAM to 16K RAM costs only $290), you can order the computer by placing a $100 deposit instead of the full amount, and (if Radio Shack's projections are correct) you'll be able to obtain disks, printers and repair service more quickly and cheaply than if you buy Commodore.

On the other hand, Commodore offers a better version of BASIC (though Radio Shack will probably catch up), a better editor and a faster CPU: Commodore's 6502 CPU is faster than Radio Shack's Z80, though the Z80 has the advantage of utilizing less memory.

But if you try to order a consumer system from Commodore or Radio Shack, you'll most likely have to wait several months before your order is filled; and during the first half of 1978, you'll have a hard time trying to add disks, a printer, and extra memory.

If you can't afford to wait, you might consider buying from the many hobby companies, which offer faster delivery, more options for expanding your system, and personal help (through computer store dealers). These companies advertise heavily in many computer publications. Browse through the ads. When reading the ads be careful. Usually the announced price does *not* include a terminal, tape recorder, BASIC, RAM and interfaces, so you'll have to pay extra. Ask the company how much. Also, ask whether the computer comes assembled or as a kit.

Minicomputers cost more than microcomputers, but offer greater speed and more software. If you'd like minicomputer software but can't afford a minicomputer, you might look into a microcomputer that imitates ("emulates") a minicomputer. Such a microcomputer (called an emulator) understands the same machine language as the minicomputer, and handles the same software, though more slowly.

Popular emulators among hobbyists include the Intercept Junior (which imitates a PDP-8), the Heathkit H-11 (which contains a microcomputer version of the PDP-11) and – somewhat more expensive – the Micronova (which imitates Data General's Nova). But since microcomputer manufacturers have been improving microcomputer software, the need for emulators has decreased.

In spite of new products, the field of personal computing is basically stagnating.

To run a business, your computer should have a fast printer, lots of memory and foolproof programs. Many microcomputer companies are starting to sell such computer systems (including the programs) for about $10,000.

During the past few months, the price of RAM memory has been dropping rapidly. Last September several companies were offering 16K RAM boards, fully assembled, for only $400. Since then, the price has dropped.

In October, S. D. Sales offered a 32K RAM board for $475 in kit form, $525 assembled. To achieve 32K, S. D. Sales used chips containing 8K bits instead of the traditional 4K bits. To obtain "8K-bit chips" cheaply, the company bought defective 16K-bit chips, and used the non-defective half of each chip. Several other companies have imitated S. D. Sales' strategy. S. D. Sales predicts the cost of *non*-defective 16K-bit chips will drop rapidly during the first quarter of 1978, whereupon the company will start selling 64K boards instead of 32K boards.

If you'd like something better than the usual Minifloppy disk drive, but can't afford a full floppy, take a look at Shugart's new intermediate model. It writes on both sides of a Minifloppy disk by using two arms; and on each track of each side. It also records at twice the normal density, so altogether, it records 4 times as much information as the standard Minifloppy drive.

In addition, Shugart's average access is faster (448 msec instead of 638) and it makes fewer errors. Called the SA450, it costs $450, whereas the traditional drive (the SA400) costs $355.

At first glance, the SA450 seems similar to the "double-sided, double-density" drive Wangco put out a year ago for only $400. But, Wangco's drive has only one arm; to write on Wangco's flip side, you must flip the disk by hand.

But Shugart's new drive won't help you unless someone invents a controller for it.

Businessmen have complained that even full floppy disks don't hold enough data to run a company; so MITS is now offering a business system that includes an Altair 8800B attached to 10 megabytes on hard disk, plus a terminal, a bi-directional printer, 64K of RAM, BASIC and an accounting package. But the total cost of the system comes to $15,950, far beyond the usual price tag for a "personal computer".

Ryal Poppa, president of PerTec, which owns MITS, says that MITS plans to shift their emphasis away from hobbyists toward businessmen.

MITS' old competitor, IMSAI, seems to be moving in the same direction – their ads show an atrractive blonde secretary sitting at an IMSAI computer – an approach more likely to appeal to businessmen than to hobbyists. And IMSAI's newest computer model has a tiny CRT screen, imitating the business-oriented $9000 IBM 5100.

Last September, MITS announced a "timesharing" version of BASIC. It lets you attach a number of terminals to your computer and run different programs simultaneously. Since timesharing BASIC costs $750, it won't appeal to hobbyists, but it might appeal to schools. On the whole, it is an extension of Disk Extended BASIC; but to prevent one user from destroying another's program, it prohibits PEEK and POKE commands. Now, several other manufacturers are offering their own versions of timesharing BASIC at widely varying prices.

Zilog will make a 16-bit CPU, called the Z-800, and Intel has similar plans. The Z-800 will be Zilog's third CPU. The others are the Z80 (which competes against Intel's 8085) and the Z8 (which includes RAM and competes against Intel's 8084). Since both Zilog and Intel are planning 16-bit CPUs, the rest of the microcomputer industry will probably follow suit, and 8-bit CPUs will become obsolete. The TMS 9900, the LSI-11 and the Micronova 16-bit microprocessors are already available to hobbyists.

Teletypes are becoming cheaper. You can get a standard ASR-33 Teletype for $845, and a KSR-33 Teletype for $600. (The KSR-33 resembles the ASR-33 but lacks the paper-tape unit.) These sample prices apply to used equipment with a 90-day warranty. Add to this cost the price of shipping in a padded van.

Many computer stores and mail-order houses have begun selling souped-up IBM Selectric typewriters as computer terminals, cheaply. But before you plunk down your money on anyone's table, make sure you're getting a completely assembled ready-to-run terminal that doesn't require hardware or software adjustment or additional accessories. Altogether, you can buy a used, reconditioned IBM Selectric typewriter complete with an ASCII/EBCDIC interface and cables with a warranty for $900.

The cost of printers has also dropped. Until recently, the most remarkable low-cost printer was the Axiom EX-800, which prints 80 characters per line, 2 lines per second, and costs only $655. Unfortunately, Axiom's characters are tiny, and the electrostatic paper is narrow (5 inches) and expensive (the Byte Shops charge $5 for a 240-foot roll); and for reliability, you need the serial-interface option, which costs another $55. But now Axiom is facing some competition.

For one, the Centronics Micro-1 is faster (3 lines per second) and cheaper ($595). Integral Data Systems offers a printer that prints on wider lines (132 characters), uses cheap, ordinary paper, includes a serial interface, and costs only $475; its only disadvantage is its slower speed (120 characters per second).

The cheapest decent printer, the Exorciser, costs just $325 and uses ordinary paper; but it's slow – 10 characters per second. You can shave about $50 off the Exorciser's price, if you don't mind not having a cover.

In spite of those new products – and the many more that space prevents me from mentioning – the field of personal computing is basically stagnating. Everyone's waiting to see whether Radio Shack and Commodore will make good on their promises to deliver, quickly and cheaply, consumer computers with disks, printers and lots of memory. Radio Shack said it would provide an interface to an S-100 bus, which would let you attach products from other manufacturers – let's wait and see.

But even if Radio Shack and Commodore pull through, I don't think you'll see a computer in every home quickly. The average American is reluctant to pay $600 for a new gadget. And the manufacturers have yet to demonstrate an advantage, other than novelty, to owning one.

At *Personal Computing*'s October Chicago show – which appeared on the local evening television news and the front page of Chicago newspapers– many of the people in attendance asked how they could apply a computer in the home and justify its expense, for use as something besides playing games, running home-grown businesses or doing tasks that could be done just as easily by hand (such as keeping a diary). I couldn't think of a convincing answer – Can you?

WHAT'S COMING UP!

Subsystems

A new instruction manual, "Using the **PERCOM CIS-30+** with the MITS 680b Computer", has been released by Percom Data Company. Percom makes the CIS-30+ unit which interfaces two cassette recorders and a data terminal to 6800-based microcomputers. The new manual now extends the adaptability of the unit to the MITS computer. The CIS-30 sells for $79.95 in kit form and $99.95 assembled. If not available from home computer dealers, the unit may be ordered directly from factory.

Designated the **PRO**, a new flexible keyboard is available from Cherry Electrical Products Corporation. The keyboard features an alpha lock key that changes outputs from typewriter to Teletype code; and five unassigned relegendable keys. The PRO is designed to piggyback a "daughter" board easily. Options available include negative logic, high voltage output, encoded or non-encoded outputs, flexible key assignments, provisions for auxiliary keyboard, automatic repeat function, optional parity bit, varied strobe pulse width, output latch with auxiliary circuits and optional shift-control mode. For a full description of the keyboard, write to Cherry for a free copy of the 8-page "Meet the PRO".

Shugart Associates announces their **SA450**, a double-sided, double-density, double-headed minifloppy drive. Brother to its SA400 minifloppy, the SA450 reads and writes data on both sides of a mini disk without removing it from the drive. This design uses two proprietary glass-bonded ferrite ceramic read/write heads based upon SA850 double-sided drive technology. The new SA450 drive, costing $450, will store up to four times the online data of the SA400, or 440 Kbytes unformatted.

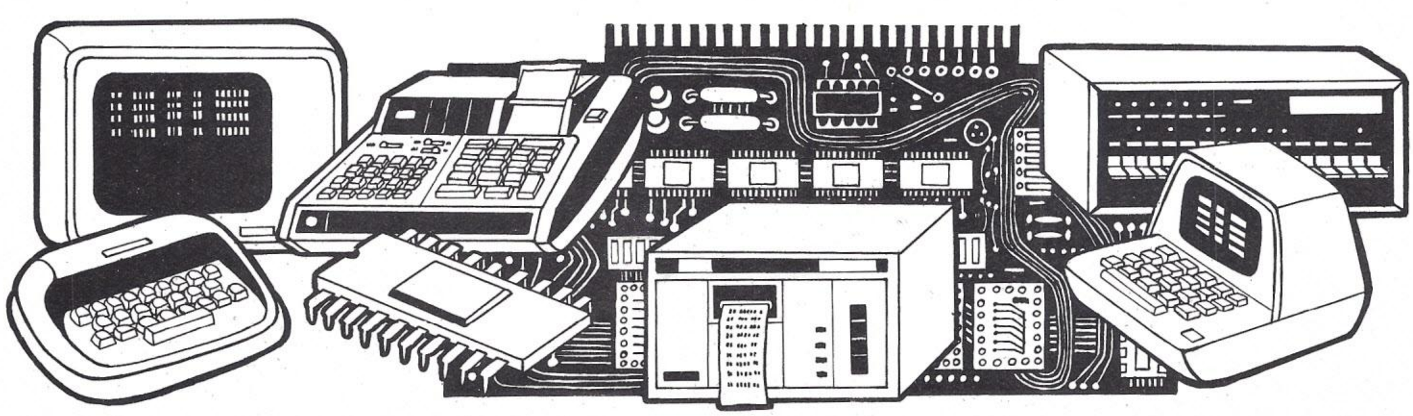
A processor board for use in S-100 bus computers is available from North Star Computers, Inc. This one is a 4 MHZ, Z80A, (**ZPB**) that operates in systems with or without front panels. It includes space for 1K of 2708 EPROM. It is available as a kit for $199 ($259 assembled). EPROM options can be obtained as a kit for $49 or, fully assembled, $69. North Star also announces availability of its new **16K RAM board** designed for use in both 8080 and Z80 computer systems. The low power board uses 200 ns dynamic RAM chips and the on-board memory refresh is invisible to the processor. The kit costs $399; assembled, $459. Addressing the board is done through switch selections in two 8K sections.

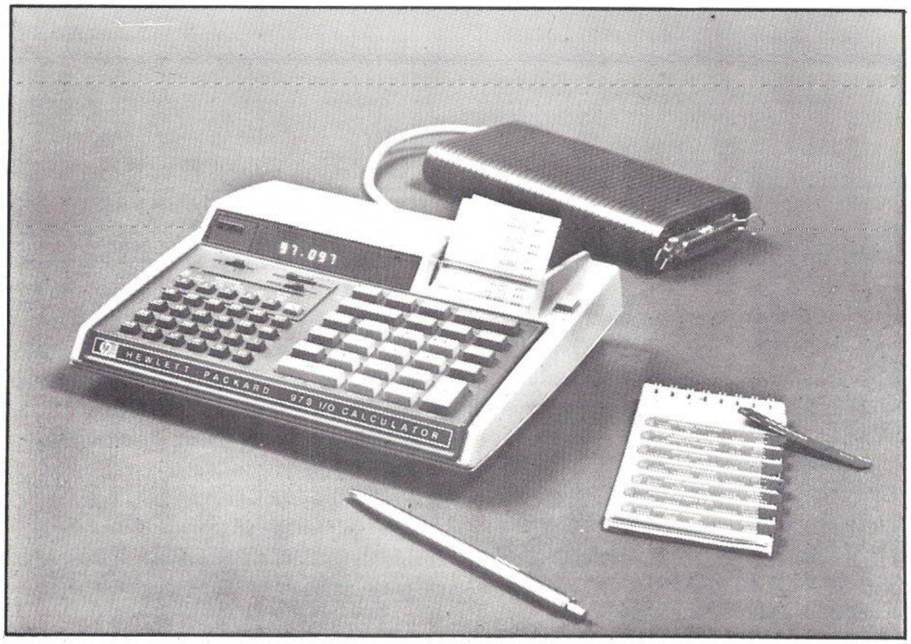
New hardware, software, components and systems – that's what's coming up here. As we receive news in these categories we condense it, pass it on to you, and include information on where to obtain more elaborate data. (See circle numbers at the tail-end of the section.)

A **VDB** (Video Display Board) that is low in cost and yet provides the capabilities of video terminals costing much more has been introduced by Technical Design Labs. Designed as a video interface for the S-100 bus microcomputers, the VDB consists of two boards, one piggybacked to the other. The unit occupies one edge connector on the bus but takes up the space of two boards. The VDB contains its own display buffer memory and provides two pages of display, each with 25 rows of 80 characters. The display accepts data at a 400,000 character per second clip.

Introl, a new unit by Mountain Hardware, controls AC devices remotely from any S-100 bus or Apple II computer over existing 110 voltage in homes, factories, schools and businesses. The Introl system impresses a code modulated 50 KHz on AC wiring, then it decodes the signal at any outlet to switch AC devices on and off. Devices that Introl can affect are lights, TVs, stereos, solenoid valves, sprinklers, burglar alarms, heating, air conditioning and other components. Programmed in BASIC or assembler language, Introl costs $149 in kit, $189 assembled. Optionals available at extra cost include Dual Channel AC Remotes ($99 for kit); and a Calendar/Clock Board ($179 in kit form). The options are also available assembled.

A logic control unit for solar space heating systems is available from Contemporary Systems, Inc. The **LCU-110** unit provides totally automated performance yet permits manual operation of the system when desired. Priced





at $584 the LCU-110 is designed to interface with conventional heating systems through another board with plug-in logic and I/O boards.

Hewlett-Packard Company has a new computing data acquisition device called the **HP Model 97S I/O** calculator. Based around its HP-97 programmable printing calculator, the new model uses BCD interfacing to gather data from a wide range of instruments. The HP 97S I/O then manipulates the data according to user-designed programs and produces a printed hardcopy report. With the HP-97S I/O calculator the user can take an instrument measurement and compare it to a standard, calibrate in data, or do computation on each individual reading. The list of products that can be used more efficiently with the HP-97S I/O is large, says Hewlett-Packard. Included in that list are electronic balances, photometers, thermal conductivity measurement devices, colorimeters, physical parameter measuring equipment, densitometers, pH meters and others. The basic system costs $1375 with at least 12 week delivery.

Dynabyte Company offers two new **static RAM modules** in either 16K or 32K and with a choice of 450 nanoseconds or 250 nanoseconds. The 250 nanosecond RAMs are compatible with 4MHz Z-80 processors. The units are all completely assembled, tested, burned-in and guaranteed for one year. The extensive testing results in a high level of reliability, making the RAM modules suitable for industrial applications and business data processing. Prices on the four models are $525 for 16K, 450 ns; $555 for the 16K, 250 ns;

$925 for the 32K, 450 ns; and $995 for the 32K, 250 ns. The modules are addressed in four separate 4K blocks along 4K boundaries and each block is "write protected". If an attempt is made to write into a protected block, audio-visual alarms are given.

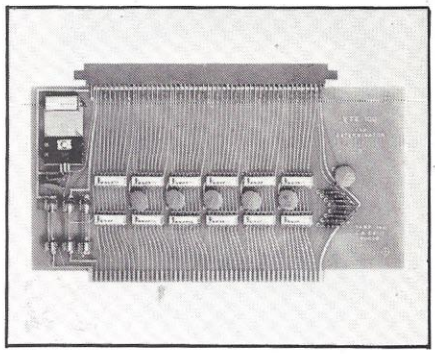
Gandalf Data, Inc., manufacturer of low-cost, high-performance data transmission equipment, has added **LDM 404B** to its line of medium distance high speed modems. The LDM 404B is designed for metropolitan areas where "unloaded" cable pairs from telephone companies are unavailable. The unit operates at 4800 bps over standard tariffed 3002 voice grade leased lines.

Gandalf also announces that it has a new **"Front End Switch"**. This device automatically switches a terminal or a number of terminals from a computer source that has failed to a back-up computer. Features of the standard Front End Switch include speeds to 9600 bps asynchronous and 19.2 kbps synchronous, status indicators for all channels, and standard EIE RS 232C interfaces.

Texas Instruments has a new **TI Programmer** calculator, available through retail outlets. The calculator is useful to programmers and others in data processing work who need to do arithmetic in three different number bases and perform conversions to and from these bases rapidly. The number bases include hexadecimal (base 16), octal (base 8) and decimal. Because most computers use a binary number system for internal storage data and addressing, programmers often need to convert coded numbers to other related number bases. Among other functions, the TI Programmer converts memory addresses to decimal form, adds relative addresses to a base address to find specific computer memory locations, or determines available space in a computer's memory. Suggested retail price for the calculator is $60.

From Germany comes a new **pocket-sized teleprinter.** Manufactured by Gleichmann & Co., this miniature printer costs only about 1/10th as much as the conventional solution, say the manufacturers. The device has a built-in microprocessor, is totally electronic and is noiseless. The keyboard has 64 alphanumeric characters, produces a serial ASCII-code and is compatible with any 20 mA current loop equipment. It has a nine-digit display and a transmission speed of 110 baud.

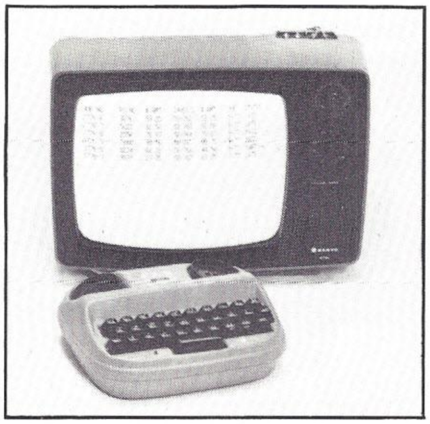
The Exterminator (VTE), from Vamp Inc., is a dual function board



which terminates an entire S-100 bus and, at the same time, serves as a card extender. Among its terminating properties are elimination of crosstalk between buses, overshoots, ringing, and scrambling of data. Functioning as a card extender, the VTE-100 eliminates

interference from adjacent boards which may radiate digital radio frequencies. Cost of the Exterminator is $49.95 (plus $2 shipping) fully assembled.

A new terminal, introduced by Micon Industries, turns any standard



television set into a low-cost time-sharing unit. The **TIGER** contains an acoustic coupler for remote time-sharing, full ASCII keyboard and TV electronics that provide interconnections to a standard TV set via antenna input. Up to 1024 characters may be simultaneously displayed in switch selectable formats for 8 or 16 lines of 32 or 64 characters per line. The TIGER terminal sells for $500.

Infinite Incorporated has a new general purpose I/O board called **MFIO-1**. Features of the unit include I/O mapped parallel input port for keyboard, selectable baud rates of 50 to 19200, compatibility with the S-100, 128 bytes of RAM and slots for EPROMs. Total power requirement is less than 1A. Prices are: assembled $282; kit, $234; bare board, $49; set of 2 ROMs, $65.95.

PUP-1 is a new microcomputer in the family of Seals Electronics. Features include a one-piece removable card cage and removable panels for quick servicing when needed. The PUP-1 has an 8-bit Z-80 CPU 2.5 MHz standard; extended BASIC; DOS; S-100 compatible; 32K memory; built-in dual Shugart mini-floppy standard, 86K per disk.

Systems

The **Noval 760** home computer comes with its own wooden desk. Keyboard, CRT, cassette tape drive and printer fold out of sight when not in use. The $3385 hardware and software system uses a Z-80 microprocessor with 32K RAM and 3K PROM. Features include 8 built-in I/O ports, power supply, 16K additional RAM, E-PROM burner, bus-extender, film reader, paper tape reader and dual hand-held keyboards for competitive games. Software includes audio tone generator, graphics, on-line debugging and BASIC on mag tape or PROM.

Compucolor Corporation plans to mass market its new **Compucolor II**, a personal computer with 8-color, 13-inch display. The computer features a typewriter-like keyboard with extra function keys, 27K memory, an an 8080A CPU and a mini-disk drive

mass storage device. Available programs include not only check book balancing, home budgeting and income tax compilation but also games with color graphics – Star Trek, Backjack, Chess, Checkers, Othello and Biorhythms.

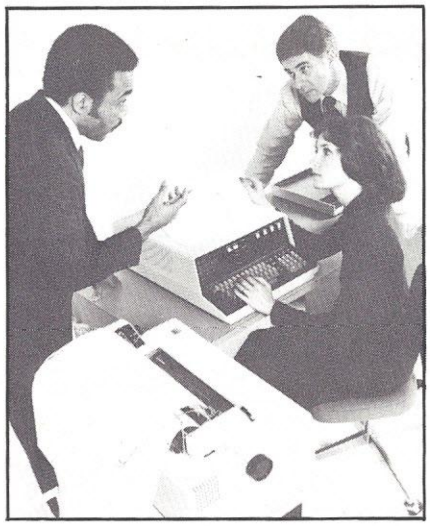
PerTec's Microsystems Division recently introduced a compact, desk-top computer in a cameo white case. Called the **Attache**, the 25-pound unit comes with full ASCII keyboard (upper and lower case). Its circuitry, built around the 8080 MPU, uses the S-100 bus with 10-slot board capability.

Retailing for $1449 assembled and tested, the basic configuration includes CPU board, video board and turnkey monitor board. Standard features include light emitting diode (LED) indicators for on/off and systems status; a reset switch which returns to PROM monitor; a monitor PROM that controls operation of the computer from the keyboard; and a video output jack (75 ohms).

For small businesses, PerTec has a complete hardware and software system called the **MITS 300**, available in two configurations. MITS 300/55, priced at $15,950, is a hard-disk system; and MITS 300/25, priced at $11,450, is a floppy disk system. Both provide word processing, inventory control and accounting functions including general ledger, accounts payable, accounts receivable and payroll.

Each configuration is comprised of a mainframe, a CRT terminal on a disk and a printer on a pedestal. The system incorporates a MITS/Altair 8800b turnkey mainframe with 64K of Dynamic RAM, 1K of PROM and a serial input/output interface. A MITS/Altair B-100 CRT terminal with 12-inch, non-glare monitor displays 24 lines of 80 characters per line with a memory page of 1920 characters. Its printer, a MITS/Altair C-700 with bi-directional operation, can print 60 characters per second, 26 lines per minute. The system also includes BASIC language software and a MITS/Altair A08 Accounting Package.

International Business Machines also has a desk-top computer for small businesses. The **IBM 5110** ranges in price from a base model at $9,875 to a fully configured system at $32,925. The computer can automate such business functions as general ledger and accounts payable. In addition, the manufacturers say, the 5110 can provide a variety of reports to help management analyze sales, schedule resources, reduce inventory cost and plan growth.



Users can display upper and lower case data anywhere on the 1024-character CRT. *(continued on following page)*

AID-80F, a disk-based computer from Mostek, provides tools for hardware/software development and debugging. The system is organized around a Z80 chip plus 16K bytes of on-board

RAM. Memory and expansion boards include 16K bytes of RAM and four eight-bit I/O ports.

Software programs and features of the $5995 system include text editor (allows editing of any size file), relocating linking loader (loads and relocates object modules in RAM), peripheral interchange program (copies files from disk to disk, disk to peripheral, or any peripheral to any other peripheral), and debugger.

Software

Technical Design Labs (TDC) has announced an **ANSI Standard FORTRAN IV** for Z80 microcomputers. Priced at $349, the disk-oriented system runs with less than 24K with a DOS and is available in both FDOS IV and CP/M versions. Several extensions, including full type conversion, full library search, full formatted I/O, sequential and direct access I/O, hex constants and control placement of data and code areas, are available.

TDL also has an upgraded Super BASIC called **Version 3.0**. The $249 package allows the user to specify special error handling routines to process programming errors without aborting the program.

Serial input and output of data comes from the Zapple Monitor defined reader and punch devices. This data may be in ASCII (using INPUT and PRINT commands) or in binary (READ and WRITE). For non-controlled reader/punch devices, a high-speed binary mode is provided (MLOAD and MSAVE).

During data input, ON EOF GO TO provides end of file detection, The VARADR function allows the address of a particular variable to be passed to an assembly language routine through the CALL statement, permitting routines to return data to the calling program.

Version 3.0, which occupies 12K of core, runs on TDL's Z80 microcomputer system, but is adaptable to other Z80-based systems.

Electronic Product Associates' **Compiler BASIC** was designed for building business applications. Features of the $250 package include decimal arithmetic for penny amounts up to $99,999,999.99, formatted output, strings and multiple disk file I/O.

Futuredata Computer Corporation has released a $300 **Universal BASIC Compiler** that runs on 8080, 8085, 6800 and Z80 microprocessors. Special

features include string variables, array variables, bit functions, PEEK, POKE, INP and OUT functions. Users can intermix assembly language instructions directly in BASIC programs. According to the manufacturers, this version of BASIC is well suited for real-time process control and test equipment.

FDOS-III, a floppy disk operating system from PerTec, has a relocatable assembler for Z80 and 8080 code. Console communications are in decimal or hex. BATCH command allows automatic chain operations, and the system includes an optional operator prompt feature for variable input requirements.

Single command operations give users disk-to-disk program editing; disk-to-punch device transfer; reader-to-disk transfer; and disk-to-disk transfer. Available commands include Copy, Alloc, Delet, Pack, Delpk, Edit, View, List, Libo, Dump, Load, Merge, Print, Renam, Run, Link and Exit. FDOS-III is contained in a 1K PROM located on a plug-in interface card.

Algorithmics' **ZAPS Cassette Operating System** contains a full Z80 assembler, text editor, in-memory file system and labeled cassette tape storage system with cyclic redundancy checks on all tape operations. The system will run on most 8080 and Z80 processors, according to Algorithmics.

Editor commands include input, insert, delete, replace, change, global change, up, down, top, bottom, string search, print and print current line number. ZAPS is available for $60 on Tarbell, Digital Group and TDL tape formats.

Designed for the secretary or non-technical user, Ohio Scientific's **Word Processor OS-WP1** runs on any of their disk-based computer systems. The manufacturer says the $79 package is useful for writing letters, manuals, reports and everyday forms found in small businesses. Internal GET and PUT file commands transfer individual files from memory to disk. Printer control commands can be used with most impact or matrix computer printer or word processing printers, manufacturers say. A formatted output mode allows users to perform left and right justification of text without line numbers at designated widths of from 20 to 70 characters.

Micro-SEED is a microcomputer database management system from Technical Design Labs. According to TDL, the system is useful in hobby, scientific and commercial applications such as letter-writing systems, accounting and inventory packages, lab data collection and analysis, online telephone directories and marketing/sales information systems. The package operates on TDL's Xitan Z80 computers and requires TDL's Z80 Disk System configuration plus an additional 48K of RAM memory.

Z-TEL, TDL's Z80 text editor, allows users to move large blocks of text inside the buffer. Other features include macro expressions, textual and numerical error messages, branching from one part of a command string to another, nested iteration and backward search. Z-TEL costs $50 on paper tape and $40 on cassettes; a disk version is scheduled.

Sunset Technologies offers a **Histogram Program** on cassette tape for $25. The program, which automatically

CIRCLE 33

draws bar and point graphs, allows up to 109 data points on the x axis and accepts data up to six digits on either side of the decimal point. Written in 8080 machine language code, the program uses less than 4K decimal bytes.

For do-it-yourself programmers, Compac has a **Microprocessor Assembly Language Coding Form** for $2.25 per pad (3 for $6) and a **Base Conversion Table** (BIN, OCT, DEC, HEX, 2's COMP) for $2 each (3 for $5).

Buyers' Guide

Subsystems

CIS-30+
Percom Data Co.
318 Barnes St.
Garland, TX 75042
Circle No. 110

PRO KEYBOARD
Cherry Electrical Products Corp.
P.O. Box 718
Waukegan, IL 60085
Circle No. 111

SA450 MINI DISK
Shugart Assoc.
415 Oakmead Parkway
Sunnyvale, CA 94086
Circle No. 112

ZPB BOARD and 16K RAM BOARD
North Star Computers, Inc.
2547-9th Street
Berkeley, CA 94710
Circle No. 113

HP MODEL 97S I/O CALCULATOR
Inquiries Manager
Hewlett-Packard Company
1507 Page Mill Road
Palo Alto, CA 94304
Circle No. 114

VDB
Technical Design Labs
Research Park, Bldg H
1101 State Road
Princeton, NJ 08540
Circle No. 115

INTROL
Mountain Hardware Inc.
PO Box 1133
Ben Lomond, CA 95005
Circle No. 116

LCU-110 SOLAR CONTROLLER
Contemporary Systems, Inc.
68 Charlonee St.
Jaffrey, NH 03452
Circle No. 117

DYNABYTE RAM MODULE
Dynabyte, Inc.
4020 Fabian
Palo Alto, CA 94303
Circle No. 118

THE EXTERMINATOR
Vamp, Inc.
PO Box 29315
Los Angeles, CA 90029
Circle No. 119

LDM 404 B MODEM and FRONT END SWITCH
Gandalf Data, Inc.
1019 South Noel
Wheeling, IL 60090
Circle No. 120

TI PROGRAMMER CALCULATOR
Texas Instruments, Inc.
Consumer Relations
PO Box 53 (Att. TIP)
Lubbock, TX 79408
Circle No. 121

POCKET SIZED TELEPRINTER
Gleichmann & Co.
Wormser Str. 9
D 6710
Frankenthal, Germany
Circle No. 122

TIGER TERMINAL
Micon Industries
252 Oak Street
Oakland, CA 94607
Circle No. 123

MFIO-1 I/O BOARD
Infinite Inc.
1924 Waverly Place
Melbourne, FL 32901
Circle No. 124

PUP-1
Seals Electronics
10728 Dutch Town Road
Concord, TN 37922
Circle No. 125

Systems

NOVAL 760
Noval Incorporated
8401 Aero Drive
San Diego, CA 92123
Circle No. 126

COMPUCOLOR II
Compucolor Corporation
PO Box 569
Norcross, GA 30091
Circle No. 127

ATTACHE and MITS 300
PerTec Computer Corporation
Microsystems Division
21111 Erwin Street
Woodland Hills, CA 92123
Circle No. 128 and 129

ATTENTION MANUFACTURERS!

Please send your product releases and news to: Products Editor, Personal Computing Magazine, 1050 Commonwealth Ave., Boston, MA 02215.

CLASSIFIEDS

CHESS IN BASIC

HERE IT IS!
A FANTASTIC CHESS GAME TO CHALLENGE BEGINNER OR EXPERT — IN STANDARD BASIC — RUNS IN 16K OF FREE MEMORY — FOR TELETYPES OR CRT'S — COMPLETE DOCUMENTATION AND A PAPER TAPE INCLUDED IN THIS SPECIAL INTRODUCTORY OFFER.

$20.00 p.p.
— MD. RESIDENTS ADD 5%

TENBERG SOFTWARE SYSTEMS
10311-J MALCOLM CIRCLE
COCKEYSVILLE, MARYLAND 21030

Personal Computing Classified Rates

Individual: $1 per word; address set free; first two words set in all capital letters for free. If you want **bold** type, add $1 for each word set bold. Add $5 if you want replies sent to us and forwarded to you.

Company: 1X: $60/inch; 3X: $55/inch; 6X: $50/inch. Display ads accepted only if camera-ready materials submitted. Agency commission paid on camera-ready ads and after the first insertion of a publisher-set ad if subsequent insertions run without changes. Publisher sets ads in format and attempts to fit copy into space requested.

Advertisement accepted, however, with understanding that space may be larger than requested. Minimum billing size one inch; ads billed in half-inch increments thereafter. Frequency rate applies to publisher-set ads only if they run without changes in subsequent issues.

To place an ad, send copy to
Classified Ad Dept.
Personal Computing
1050 Commonwealth Ave.
Boston, MA 02215

TAXES Software: Computes Form 1040 & 5 Schedules. See if you can itemize, income average, etc. Basic; 10K min. Tarbell, Sol, Northstar, CP/M, call about others. Cassette, $20, Disk, $26.

DICOMP
2053 Manzanita Dr.
Oakland, CA 94611
(415) 434-4767

Science Writing Degree Program at Boston University

Involving the study of matter and energy on the most fundamental level, the laws of physics underlie all sciences and form the basis of the technology that is part of our daily lives. Any person who wishes to understand the world and the culture in which we live has to have some understanding of physics. For this purpose, Boston University's Department of Physics has always offered a range of general interest courses. Our program has traditionally been a fine undergraduate prelude to a wide variety of careers in computer science, law, medicine, ecology, the environment, publishing, business, engineering and research. Now, starting in September, the Physics Department has added a new aspect: how to write about these diverse subjects. The new program presents students with the background required to understand different developments in the different fields of science and how to communicate effectively about them via print or electronic media. The five-year combined B.S./M.S. program is open to freshmen or to transfer students who have completed two years in physics or science at a recognized university. For more information, circle number 99 on the reader service card or direct your inquiry to:

Professor George Zimmerman
Department of Physics
Boston University
111 Cummington Street
Boston, MA 02215

What do you have in common with a plumbing contractor, an auto parts store owner and a real estate agent?

The program you designed to help your business.

That's right. Your business application could be just the solution someone else needs for their business problems. And *Personal Computing* is looking for business applications of all types – whether tailored to your own company needs or applicable to any business. Of course, the more people who can use your program, the greater your chance of seeing it in print.

But we can't publish it until you submit it. So why not send us your program? Be sure to include your program listing, program description, and sample run. (See "How to Write for *Personal Computing*" page 96.)

And we're not just interested in business applications. Send us your home and personal management, educational or recreational programs – in fact, send us any program you've written that helps you solve a problem or have fun with your computer.

Your program may help others; in return they may help you by improving on your work. And, if we publish your piece, we'll send you a check to beef up your bank account – and add to your nest egg for that extra component.

So, mail your material to: Applications Editor, *Personal Computing*, 1050 Commonwealth Ave., Boston, MA 02215, or call us at (617) 232-5470.

We look forward to hearing from you.

Systems – continued

IBM 5110
International Business Machines Corp.
General Systems Division
P.O. Box C-1645
Atlanta, GA 30301
Circle No. 130

AID-80F
Mostek Corporation
1215 W. Crosby Road
Carrollton, TX 75006
Circle No. 131

Software

ANSI STANDARD FORTRAN IV, VERSION 3.0 SUPER BASIC, MICRO-SEED, Z-TEL
Technical Design Labs
Research Park, Bldg. H
1101 State Road
Princeton, NJ 08540
Circle No. 132, 133, 134, 135

COMPILER BASIC
Electronic Product Associates
1157 Vega Street
San Diego, CA 92110
Circle No. 136

UNIVERSAL BASIC COMPILER
Futuredata Computer Corporation
11205 La Cienega Boulevard
Los Angeles, CA 90045
Circle No. 137

FDOS-III
PerTec Computer Corporation
Microsystems Division
21111 Erwin Street
Woodland Hills, CA 91367
Circle No. 138

ZAPS CASSETTE OPERATING SYSTEM
Algorithmics, Inc.
Box 56
Newton Upper Falls, MA 02164
Circle No. 139

WORD PROCESSOR OS-WP1
Ohio Scientific
Hiram, OH 44234
Circle No. 140

HISTOGRAM PROGRAM
Sunset Technologies
210 A East Ortega Street
Santa Barbara, CA 93103
Circle No. 141

MICROPROCESSOR ASSEMBLY LANGUAGE CODING FORM, BASE CONVERSION TABLE
Compac
PO Box 18470
Cleveland, OH 44118
Circle No. 142, 143

The 2nd Annual

MIDWEST PERSONAL COMPUTING EXPOSITION

October 5, 6, 7, 8, 1978 • EXPOCENTER / Chicago (Adjacent to the Merchandise Mart)

A Larger Show in 1978 Because of Its Proven Value To Individual Computer Enthusiasts

They came in droves from all sectors of society for the first show last year — 12,565 to be exact. Doctors, lawyers, business persons, computer professionals, students and home users . . . proving the modern-day marvel of personal computing is within the grasp of everyone. And that was just for starters.

Now the Show That Captured the Fascination of the Entire Midwest Moves to a Larger Facility Permitting —

- An expected 200 exhibits showing the latest computer equipment, peripherals, accessories and software by America's leading manufacturers.
- An expanded seminar program of real value for anyone eager to learn new developments and the many innovative and practical applications of personal computing . . . delivered by industry professionals.
- Easy access from all public transportation and expressways with ample parking available.

It's the Most Complete and Diversified Exhibition of Personal Computing Technology Ever Staged for the Midwest Public . . . Plan Now to Participate as Visitor, Lecturer or Exhibitor

The ONLY Personal Computing Show in Chicago Officially Sponsored by PERSONAL COMPUTING Magazine

Organized in 1978 by:

Industrial and Scientific Conference Management, Inc. — a respected leader in expositions and conferences, worldwide

222 West Adams St., Chicago, IL 60606
(312) 263-4866

Return for Complete Information to:
MIDWEST PERSONAL COMPUTING EXPOSITION
c/o Industrial and Scientific Conference Management, Inc.
222 West Adams St., Chicago, IL 60606 • (312) 263-4866

☐ I am interested in ATTENDING. Please send registration information.

☐ I am interested in EXHIBITING. Please send contract and full details.

Name ______________________ Title ____________

Company ____________________ Phone ____________

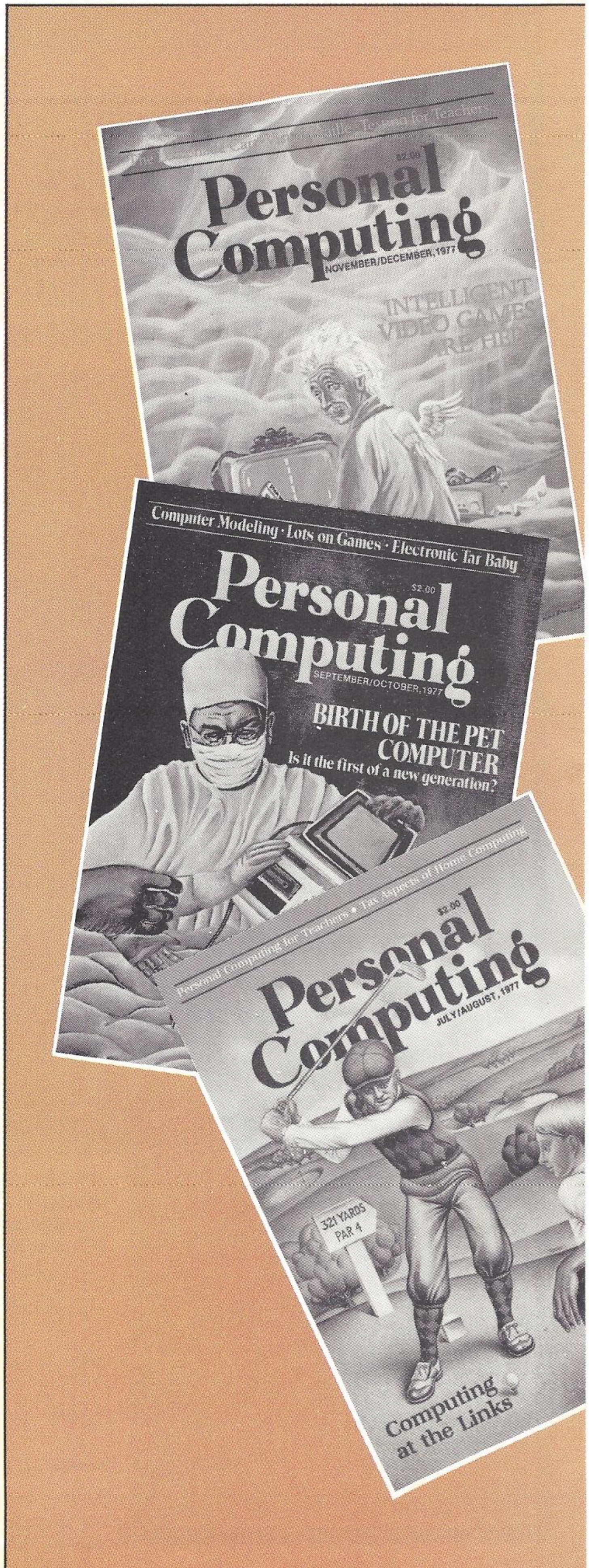
Address ______________________________________

City ______________ State ________ Zip ________

ADVERTISERS' INDEX

SALES OFFICES

Northeast: George Palken, 1050 Commonwealth Ave., Boston, MA 02215; (617) 232-5470 • **Mid-Atlantic**: Arthur Daks, 104 So. Central Ave., Valley Stream, LI, NY 11580; (516) 872-9550 • **Mid-West**: Ralph Petersen, 1 Wheaton Ctr. No. 1706, Wheaton, IL 60187; (312) 653-2480 • **Southeast**: Harold Short, Parker & Short, 1 Lakeside Office Park, Wakefield, MA 01880; (617) 246-2293 • **West**: Yuri Spiro & Carol Stagg, 823 Enchanted Way, Pacific Palisades, CA 90272; (213) 454-0624 • **Japan**: Hiro H. Irie, International Business Corp., 11-8, Narita-Higashi 1-chome, Suginami-KU, Tokyo 166; Phone: (03) 311-1746.



How to Write for Personal Computing

Have you programmed your computer to converse in Gaelic? to do your home-ec homework? to read a bedtime story to the kids? Are you a frustrated fiction writer who's caught the computer bug? Or, have you found the ideal system or the absolutely worst combination of components?

Why not share your experiences with our readers? Yes, you too can write for *Personal Computing.* You choose the topic, *any* topic. If your topic relates to computers, great. If it relates to personal computers, even better. Computer hobbyists are looking for an excuse, any excuse, to buy a computer, and you might just offer the justification they're looking for.

We accept articles for all our sections – *Launching Pad* (our tutorial section for beginners), *On the Lighter Side* (where we print humorous applications), *In the Money* (how to use your computer to benefit financially), *Digging In* (for our more "advanced" topics), and *Once Upon a Time* (where we let your imagination run wild). We'd love to see some comparisons of computers or computer products. Tell us the good *and* bad of your system.

Keep your writing simple. No, our readers are not simpletons or beginners, but if you can explain something in simple words, do so. Don't clutter your piece with unnecessary jargon. If you're already into computers, give the newcomers a hand and let them in on some of the tricks of the trade – in simple terms. Examples, analogies, and charts and diagrams help both the beginner and the more advanced user appreciate what you're saying. Feel free to use "I" and "you" to make your article more personal and meaningful to the reader. Put the reader in the position of programmer ("you"). Also, please, please do not write your entire article in caps. And please indent for each paragraph.

Some things to note. Make sure your details are accurate – especially prices, other numerical information, and company names. Don't rely on hearsay or memory.

If you write about a program you've invented, try this order (to make sure you cover all angles): state the program's purpose; show a sample run; explain what the input options are, and what the output means; show another sample run; explain the underlying theory (if any); state the language, version, and computer you used and their peculiarities; show the listing; explain the program's over-all structure; analyze the program's details line by line; and suggest how the reader might improve or change the program.

Whatever your area of interest, you can turn it into an article. For example, if you're interested in watching birds then why not try an article on how to use a computer to track bird migrations? Or if your business is ________________, why not try a piece on computers and how they can be used to ________ ________________. We're open to ideas . . . .

If you've never written for a publication before and you'd like to discuss your piece with us before beginning it, give us a call. (Please do *not* mail us vague story proposals or outlines. We'd rather see the first few paragraphs of your article.) We'd be glad to discuss what you have in mind, and offer a few ideas of our own.

As a matter of form, we prefer (and are more likely to accept) articles that have been typed. Most of our articles run around 2-4 magazine pages. (There's about 3-3/4 typewritten pages to a magazine page.) But if what you have is good, we'll compromise and print anything from a sentence to your version of the encyclopedia.

Now here's the good part: we pay for any original material we print, although the price varies depending on the *quality* of the article. (So make it good!)

Why not give it a whirl? There may be a latent Hemingway, Fitzgerald or Asimov beneath that Einsteinian exterior.

Extensys Corporation announces the EX3000 Computer Systems. This complete series of computer systems embody architectural concepts found, before now, only in costly large scale computers. The primary concept that is used through all of the EX3000 computer systems is distributed processing. The EX3000 systems distribute system processing functions to those system components that are best suited to perform those tasks. This is a state-of-the-art technology which was prohibitive until the recent introduction of cost effective computational and control microprocessors, along with programmable peripheral circuits to perform preassigned tasks. Coupling these cost effective products with the distributed processing techniques proven by many large mainframe computer manufacturers, Extensys Corporation has been able to provide extremely powerful and flexible systems. Through the efficient use of system components and their modular structure, the EX3000 Computer Systems allow configurations that meet a wide variety of particular application needs as well as offering expansion capability to satisfy increasing usage demands.

For more detail information on the EX3000 Computer Systems and how they can satisfy your processing needs, give us a call.

extensys corporation
380 Bernardo Avenue
Mountain View, CA
94040
(415) 969-6100

CIRCLE 3